

STATE ESTIMATION, SYSTEM IDENTIFICATION AND ADAPTIVE CONTROL FOR NETWORKED SYSTEMS

A Thesis Submitted to the
College of Graduate Studies and Research
in Partial Fulfillment of the Requirements
for the degree of Master of Science
in the Department of Mechanical Engineering
University of Saskatchewan
Saskatoon

By
Huazhen Fang

©Huazhen Fang, May 2009. All rights reserved.

PERMISSION TO USE

In presenting this thesis in partial fulfilment of the requirements for a Postgraduate degree from the University of Saskatchewan, I agree that the Libraries of this University may make it freely available for inspection. I further agree that permission for copying of this thesis in any manner, in whole or in part, for scholarly purposes may be granted by the professor or professors who supervised my thesis work or, in their absence, by the Head of the Department or the Dean of the College in which my thesis work was done. It is understood that any copying or publication or use of this thesis or parts thereof for financial gain shall not be allowed without my written permission. It is also understood that due recognition shall be given to me and to the University of Saskatchewan in any scholarly use which may be made of any material in my thesis.

Requests for permission to copy or to make other use of material in this thesis in whole or part should be addressed to:

Head of the Department of Mechanical Engineering
57 Campus Drive
University of Saskatchewan
Saskatoon, Saskatchewan
Canada
S7N 5A9

ABSTRACT

A networked control system (NCS) is a feedback control system that has its control loop physically connected via real-time communication networks. To meet the demands of ‘teleautomation’, modularity, integrated diagnostics, quick maintenance and decentralization of control, NCSs have received remarkable attention worldwide during the past decade. Yet despite their distinct advantages, NCSs are suffering from network-induced constraints such as time delays and packet dropouts, which may degrade system performance. Therefore, the network-induced constraints should be incorporated into the control design and related studies.

For the problem of state estimation in a network environment, we present the strategy of simultaneous input and state estimation to compensate for the effects of unknown input missing. A sub-optimal algorithm is proposed, and the stability properties are proven by analyzing the solution of a Riccati-like equation.

Despite its importance, system identification in a network environment has been studied poorly before. To identify the parameters of a system in a network environment, we modify the classical Kalman filter to obtain an algorithm that is capable of handling missing output data caused by the network medium. Convergence properties of the algorithm are established under the stochastic framework.

We further develop an adaptive control scheme for networked systems. By employing the proposed output estimator and parameter estimator, the designed adaptive control can track the expected signal. Rigorous convergence analysis of the scheme is performed under the stochastic framework as well.

ACKNOWLEDGEMENTS

I would like to thank Dr. Yang Shi first. As the supervisor, Dr. Shi provided me invaluable advisement, incomparable opportunities and great freedom in research; as a trusted friend, he always gave me timely and sincere suggestions whenever I faced difficulties in both research and life. I would give my grateful thanks to Dr. Shi for his encouraging guidance, persistent inspiration and extraordinary kindness, though words of praise are hardly enough to express my gratitude sufficiently.

I would like to thank the thesis committee members, Prof. Daniel Chen and Prof. Aryan Saadat Mehr for their constructive comments. Special thanks go to Prof. Fang-Xiang Wu for his help when I was in need.

I have had the pleasure of meeting countless talented individuals while at the University of Saskatchewan. I would like to thank my groupmates, Yang Lin, Bo Yu, Hui Zhang, Jian Wu, Ji Huang, Qiao Zhang, Lili Han and Jie Ding. With them, research becomes enjoyable adventures and life is full of sunshine.

I thank the University of Saskatchewan for her intensive academic atmosphere, nice faculty, staff and students, and of course, her financial support.

Finally, this thesis is dedicated to my parents. I could not have completed it without their unselfish love and support.

To my parents.

CONTENTS

Permission to Use	i
Abstract	ii
Acknowledgements	iii
Contents	v
List of Tables	vii
List of Figures	viii
List of Abbreviations	x
1 Introduction	1
1.1 Background	1
1.2 Previous Work	3
1.3 Contributions and Outline	4
1.3.1 Objectives and Contributions	4
1.3.2 Outline	5
2 State Estimation for Networked Systems	7
2.1 Introduction and Literature Review	7
2.2 Problem Formulation	9
2.3 Derivation of the Algorithm	10
2.3.1 Preliminaries	10
2.3.2 Input Estimation	11
2.3.3 State Estimation	13
2.3.4 Algorithm Summary	15
2.4 Convergence Analysis	16
2.5 Numerical Examples	21
2.6 Summary	24
3 System Identification for Networked Systems	30
3.1 Introduction and Literature Review	30
3.2 Problem Formulation	33
3.3 Derivation of the Algorithm	34
3.4 Convergence Analysis	36
3.4.1 Preliminaries	36
3.4.2 Convergence of Parameter Estimation	38
3.4.3 Convergence of Output Estimation	44
3.5 Numerical Examples	45
3.6 Summary	53
4 Adaptive Control for Networked Systems	54

4.1	Introduction and Literature Review	54
4.2	Problem Formulation	56
4.3	Derivation of the Algorithm	58
4.3.1	Parameter Estimation and Missing Output Estimation	58
4.3.2	Adaptive Control Design	60
4.4	Convergence Analysis	60
4.4.1	Preliminaries	61
4.4.2	Convergence Analysis	62
4.5	Numerical Examples	66
4.6	Summary	68
5	Conclusions and Future Work	72
5.1	Conclusions	72
5.2	Future Work	73
	References	75

LIST OF TABLES

2.1	Example 1: Comparisons between the norms of input and state estimation errors by Algorithm 1 and the Gillijns-Moor algorithm.	21
2.2	Example 2: Comparisons between the norms of input and state estimation errors by Algorithm 1 and the Gillijns-Moor algorithm.	23
3.1	Example 1: Intermediate parameter estimates and estimation errors ($\lambda = 0.8$ and $\gamma = 0.7$).	46
3.2	Example 2: Intermediate parameter estimates and estimation errors ($\lambda = 0.4$ and $\gamma = 0.2$).	51

LIST OF FIGURES

1.1	A networked control system.	2
2.1	Blockdiagram of simultaneous input and state estimation.	8
2.2	Example 1: Results for input and state estimation. (a) The input u_k and its estimate \hat{u}_k . (b) The first state x_{1k} and its estimate \hat{x}_{1k} . (c) The second state x_{2k} and its estimate \hat{x}_{2k}	25
2.3	Example 1: Trace of P_k^x vs. k	26
2.4	Example 2: Results for input and state estimation. (a) The input u_k and its estimate \hat{u}_k . (b) The first estimate x_{1k} and its estimate \hat{x}_{1k} . (c) The second x_{2k} and its estimates \hat{x}_{2k}	27
2.5	Example 2: Trace of P_k^x vs. k	28
2.6	Example 3: Results for input and state estimation. (a) The input u_k and its estimate \hat{u}_k . (b) The first state x_{1k} and its estimate \hat{x}_{1k} . (c) The second state x_{2k} and its estimate \hat{x}_{2k}	29
3.1	Scenario of identification over a lossy network under TCP-like protocols. . .	31
3.2	Example 1: Estimates of the parameter unknowns a_1 , a_2 , b_0 and b_1 ($\lambda = 0.8$ and $\gamma = 0.7$).	47
3.3	Example 1: Relative parameter estimation error versus time ($\lambda = 0.8$ and $\gamma = 0.7$).	47
3.4	Example 1: Comparison between estimated and true outputs ($\lambda = 0.8$ and $\gamma = 0.7$).	48
3.5	Example 1: Average output estimation error versus time ($\lambda = 0.8$ and $\gamma = 0.7$).	48
3.6	Example 2: Estimates of the parameter unknowns a_1 , a_2 , b_0 and b_1 ($\lambda = 0.4$ and $\gamma = 0.2$).	49
3.7	Example 2: Relative parameter estimation error versus time ($\lambda = 0.4$ and $\gamma = 0.2$).	49
3.8	Example 2: Comparison between estimated and true outputs ($\lambda = 0.4$ and $\gamma = 0.2$).	50
3.9	Example 2: Average output estimation error versus time ($\lambda = 0.4$ and $\gamma = 0.2$).	50
3.10	Example 3: Relative parameter estimation errors for two cases with different data missing patterns ($\lambda = 0.8$ and $\gamma = 0.7$).	52
3.11	Example 3: Average output estimation errors for two cases with different data missing patterns ($\lambda = 0.8$ and $\gamma = 0.7$).	52
4.1	An NCS with randomly missing outputs.	56
4.2	Adaptive control diagram.	60
4.3	Example 1: Output response when $\gamma = 0.85$	68
4.4	Comparison of relative parameter estimation errors for Example 1 and Example 2: Blue solid line for Example 1; red dashed line for Example 2. . . .	69
4.5	Example 1: Comparison between estimated and true outputs when $\gamma = 0.85$ (The dashed line represents output missing).	70
4.6	Example 2: Output response.	70

4.7	Example 2: Comparison between estimated and true outputs when $\gamma = 0.65$ (The dashed line represents output missing).	71
4.8	Example 3: Output response subject to parameter variation: At time $k =$ 2500, all parameters are increased by 50%.	71

LIST OF ABBREVIATIONS

AR	Autoregressive
ARX	Autoregressive Exogenous
DSP	Digital Signal Processing
EM	Expectation-Maximization
LQG	Linear Quadratic Gaussian
LQR	Linear Quadratic Regulator
LS	Least Squares
LTI	Linear Time-Invariant
MCA	Minimum Component Analysis
MIMO	Multi-Input Multi-Output
ML	Maximum Likelihood
MMSE	Minimum Mean Square Error
MPC	Model Predictive Control
MV	Minimum Variance
NCS	Networked Control System
OE	Output-Error
PID	Proportional-Integral-Derivative
RLS	Recursive Least Squares
SG	Stochastic Gradient
SISO	Single-Input Single-Output
SISE	Simultaneous Input and State Estimation
TCP	Transmission Control Protocol
UDP	User Datagram Protocol

CHAPTER 1

INTRODUCTION

1.1 Background

A networked control system (NCS), namely, is a feedback control system wherein the control loops are physically connected via real-time communication networks. Thus its fundamental feature is that the information (reference input, control input, plant output, etc) among control system components (sensors, controller, actuators, etc.) are exchanged in the form of data packets through a network. Fig. 1.1 illustrates a typical single-loop NCS setup. As the most promising application of networking technology in the control discipline, NCSs have received considerable attention worldwide during the past few years. This eye-catching trend is largely driven by three forces:

- **Engineering needs:** For engineering purposes, many industrial applications increasingly demand ‘teleautomation’, modularity, reduced complexity, integrated diagnostics, quick maintenance and decentralization of control. Focusing on the improvement of control performance, traditional control methodologies cannot meet the new demands sufficiently. However, NCSs provide a promising solution. In an NCS, the network expands the reach of controllers. Moreover, the resultant modular design makes diagnostics and maintenance easier to implement. A branch of NCSs, like distributed control systems, are to achieve decentralized control.
- **Technology progress:** Advances of related technologies such as digital signal processing (DSP), embedded computing and data networks have paved the way for the development of NCSs. The past decades witness the vast progress of DSP hardware and software. At the same time, processors become more reliable and have much higher speed, and communication networks (wired, wireless or hybrid) sweep across the world. These technologies are almost mature, being low-cost and easily available.

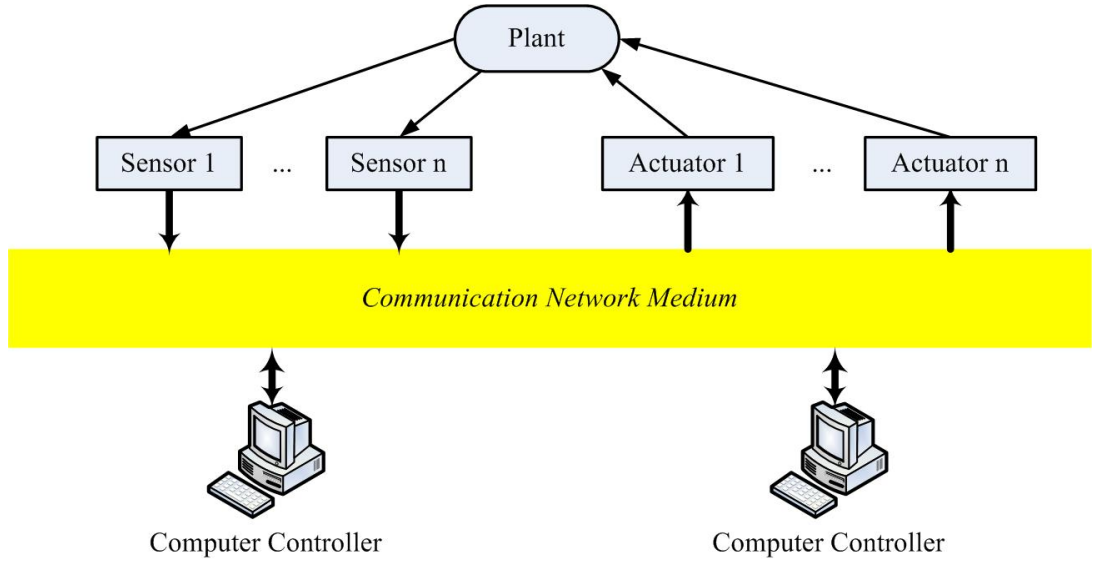


Figure 1.1: A networked control system.

- **Research importance:** Having significant implications for related research, the studies of NCSs have been placed emphasis on by the academia. The reason is obvious: NCSs form an interdisciplinary slant on automatic control, and thus provide an exceptional platform to fuse and stimulate cutting-edge research on almost every aspect of control. So far, the development of NCSs has involved robust filtering, proportional-integral-derivative (PID) control, robust control, adaptive control, control of time-delay systems, control with communication constraints and limited information, and so on.

As shown in Fig. 1.1, the plant in an NCS is installed at the remote end, whereas the computer-based controller is at the near end, and the link between them is a shared network. The output of the plant is measured by the sensors and encapsulated into data packets. The network transfers the packets to the controller, which then generates control decisions. The control input packets are sent to the actuator to drive the plant via the network. Before used by the actuator, the discrete packets are converted into continuous signals.

Although it reduces cost, wiring and maintenance complexities in practical operations, the introduction of a network causes some new issues in control designs:

- **Time delays:** Different from the ‘perfect channel’ in traditional control systems, the network is time-sharing. Therefore, data packets must wait in order to be accepted

by the network, and the time spent waiting is called ‘network access delay’. Another two kinds of delays are ‘transmission delay’ and ‘computation delay’, which happen during packet transferring over network and at the encoder, controller and decoder, respectively.

- **Packet dropouts:** Data packets may be dropped out before reaching the destinations. The possible reasons are errors of physical network link, network congestion, buffer overflows or long transmission delays. Besides, due to scheduling problems, old data packets occasionally arrive later than new ones, and thus be discarded.
- **Limited data rate:** Packets are actually a finite number of encoded bits, but the network can only carry a limited amount of information per unit time. Thus a trade-off must be made: Given a fixed data, one can use fewer-bit packets and increase the number of packets per time, and vice versa.

It is noted that the issues as a whole often result in serious degradation of system performance. Interestingly, they also make the driving force that motivates continuing progress in this research filed.

1.2 Previous Work

As just mentioned, control engineers invariably confront three notable issues in practical designs of NCSs. Current research, consequently, has been focused on developing control related subjects under such issues. The existing research on NCSs mainly targets at three problems: state estimation, stability analysis and stabilization and controller synthesis under network environment. We will review a number of the most representative works on NCSs. For a fuller listing of works in the NCS realm, the reader is referred to the bibliography and the references therein. The works closely pertinent to research in the thesis will be separately surveyed in detail in each of the following chapters.

- **Networked state estimation:** For a plant in a loop built upon lossy networks, state estimation is complicated. Mostly, a one-side link is assumed from the plant to a remote estimator, and a Bernoulli process is used to describe the random packet dropouts. The classical Kalman filter can be modified to adapt to the network environment [68; 69]. A basic principle in Kalman filtering is the minimum error

variance design, which is also widely used in solving the considered problem [22; 61; 62]. Robust \mathcal{H}_∞ filtering is another promising way and some works are dedicated to this topic [25; 43].

- **Stability and stabilization:** Time delays and packet dropouts highly degrade the performance or even destroy the stability of NCSs. Various methods have thus been proposed to deal with stability conditions or stabilization. In [89; 91; 93], random time delays are modeled as Markov chain processes, but different types of stability and related stabilization problems are studied, respectively. For an NCS, input-state and input-output stability are often concerned and discussed a lot [59; 60; 72]. From the view point of the controller design, stabilization is an important task. A few techniques, including state or output feedback control, model predictive control (MPC) have been applied [26; 54; 81].
- **Controller synthesis:** An NCS is to be controlled ultimately, so the controller design is of prime interest to researchers. Design of PID, linear quadratic Gaussian (LQG), MPC and robust \mathcal{H}_∞ controllers have been developed for NCSs [46; 55; 70; 84; 87; 92]. A novel trend on this aspect is to combine controller design and network scheduling to get an optimal integrated solution [24].

1.3 Contributions and Outline

1.3.1 Objectives and Contributions

Although NCSs have been studied widely, many problems have not been fully investigated and still remain challenging. We propose the following research topics in the thesis:

- **State estimation with unknown input:** State estimation is useful or even indispensable in helping understand the behavior of a system and developing control schemes. Most previous research on state estimation over lossy networks, especially wireless sensor networks, uses a state-space model without considering external inputs. Yet in NCSs, not only is the input unignorable, the input missing is unknown to us. This imposes intrinsic barriers to state estimation for NCSs. We would approach the problem by simultaneous state and input estimation to reduce the effects of unknown input missing to the minimum.

- **System identification under network environment:** The fact that modern control usually requires reliable models establishes the fundamental importance of system identification. A little surprisingly, networked system identification is a new problem in NCS research and has not been extensively studied yet. The challenge to be confronted with is missing data. Since classical system identification approaches fail if applied directly, we seek to develop novel effective methods to solve the problem.
- **Adaptive control under network environment:** Due to the uncertain nature of networks, NCSs are subject to changing running conditions. Therefore, a plausible question is how to design a control law that adapts itself to the network environment. This consideration leads us to study adaptive control schemes for NCSs, on which few current research has been carried out. In the thesis, we would present a model reference adaptive control scheme based on results of networked system identification.

The main contributions of the thesis can be summarized briefly as follows:

- We explore three significant problems in the field of NCSs, i.e., state estimation, system identification and adaptive control.
- We derive a solution for each problem, and analyze related properties rigorously to ensure effectiveness.
- We evaluate the performance of the solutions by a large number of simulations.

It is worth noting that the studies of NCSs involve much complexity and are at the budding age. Thus the thesis is a preliminary adventure of the wide world of NCSs. In addition, out of research needs, different NCS structures are to be considered for different problems.

1.3.2 Outline

In Chapter 2, we investigate simultaneous input and state estimation for linear discrete-time systems. The problem is considered in the setting of NCSs, in which information regarding input missing is not available. We develop a sub-optimal joint estimation algorithm. We further show that the stability and reliability of the proposed algorithm can be guaranteed under certain conditions by analyzing a Riccati-like equation. The algorithm's performance is illustrated by numerical examples.

In Chapter 3, we consider the problem of parameter estimation and output estimation for systems in a network environment based on Transmission Control Protocol (TCP). Based on the incomplete data caused by network-induced packet dropouts, we develop a recursive algorithm for parameter estimation by modifying the classical Kalman filter based algorithm. Under the stochastic framework, convergence properties of the algorithm are established. Simulation results verify the effectiveness of the proposed algorithm.

In Chapter 4, we consider the problem of adaptive control for NCSs with unknown model parameters and randomly missing outputs. In particular, for a system with the output-error (OE) model placed in a network environment, the randomly missing output feature is modeled as a Bernoulli process. Then an adaptive control is designed to make the output track the desired signal. Convergence properties of the proposed algorithms are analyzed in detail. We also show the effectiveness of the proposed method by simulation examples.

In Chapter 5, we summarize and draw some concluding remarks from the thesis research. Suggestions for some future work in the field of NCSs are presented as well in this chapter.

The notation used throughout the thesis is fairly standard. Small case letters denote vectors and capital letters denote matrices. For matrices and vectors, the superscript ‘T’ indicates transpose. We use $\det(X)$ to indicate the determinant of square matrix X . For symmetric matrix, $X > 0$ or $X \geq 0$ indicates that X is positive definite or nonnegative definite, respectively, and $X > Y$ indicates $X - Y > 0$. We use $\|\cdot\|_2$ and $\|\cdot\|_\infty$ to denote the 2-norm and ∞ -norm, respectively. ‘E’ denotes the expectation. $\lambda_{\max/\min}(X)$ represents the maximum/minimum eigenvalue of X ; $|X| = \det(X)$ is the determinant of a square matrix X ; $\|X\|^2 = \text{tr}(XX^T)$ stands for the trace of XX^T . If $\exists \delta_0 \in \mathbb{R}^+$ and $k_0 \in \mathbb{Z}^+$, $|f(k)| \leq \delta_0 g(k)$ for $k \geq k_0$, then $f(k) = O(g(k))$; if $f(k)/g(k) \rightarrow 0$ for $k \rightarrow \infty$, then $f(k) = o(g(k))$.

CHAPTER 2

STATE ESTIMATION FOR NETWORKED SYSTEMS

2.1 Introduction and Literature Review

State estimation is of much importance in networked control based applications such as remote sensing, telerobotics and sensor networks. A major difficulty while handling the problem is: Both control input and plant output are subject to missing. The plant output missing is known, whereas the control input missing is beyond our observation [90]. As a consequence, the problem is rather challenging and has been studied poorly. In our research, we attempt to solve the problem partially, that is, we will study a simplified problem. We assume that the input is totally unknown but all the output is available for use. Then we study simultaneous input and state estimation (SISE). The problem scenario is illustrated in Fig. 2.1. The simplification to SISE is a little idealistic, but serves appropriately as a preliminary attack at the original problem.

Not limited to NCSs, the SISE problem for dynamic systems actually has a wide range of applications, such as fault detection and diagnosis [63], maneuvering target tracking [53], geophysics and environmentology [67], where inputs are often unmeasurable or inaccessible. Due to its practical significance, this problem has received considerable attention during the past several decades.

For different applications, related SISE research in the existing literature can be mainly classified into three types:

- **State estimation subject to unknown inputs:** An unbiased minimum-variance linear state filter is developed in [49], in which the state estimation is designed independently with the unknown inputs. The design has been extended to a more general filter structure in [13] and the convergence conditions are also given for the time-invariant case. Further, in [14] the same problem is considered for a system with direct feedthrough. It provide an optimal filter design and stability conditions. The

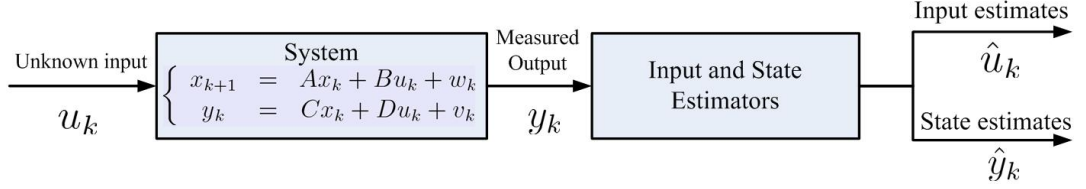


Figure 2.1: Blockdiagram of simultaneous input and state estimation.

optimization problem of the recursive filter in [49] and [13] is to minimize the error variance subject to unbiasedness constraints; however, the optimality of the filter is not proven. The proof of optimality is given in [48]. Without using optimization techniques, an alternative method to design state observers with unknown inputs is through matrix calculations, for example, see [83; 15; 71]. Sliding mode observer is another promising way to estimate states of a system subject to unknown inputs. In [23], a sliding mode observer is proposed and the convergence of the observer is proven either asymptotical or in finite time.

- **Unknown input estimation:** In many practical applications, such as fault detection and diagnosis, it is appealing to determine unknown inputs of a dynamic system. There exist numerous works in this area, for example, see [36; 78; 21; 82] and the references therein.
- **Simultaneous input and state estimation:** The above two categories focus on the estimation of either unknown inputs or system states, but not at the same time. ‘Killing two birds with one stone’, SISE has become very attractive in recent years, which is challenging since state and input estimations are inherently interconnected and coupled. In [42], a two-stage Kalman filter and an input filtering technique are combined to achieve joint estimation. An asymptotic input and state estimation scheme is proposed in [12] for a class of uncertain systems with some assumptions on the systems. Employing an LMI based technique, an SISE approach is presented in [33] for multi-input multi-output systems subject to Lipschitz nonlinearity. In [29] and [30], a set of multi-step recursive filters is proposed to jointly estimate inputs and states by minimizing the error variance for discrete-time linear systems without and with the direct feedthrough, respectively. The obtained algorithms are proven to be featured by optimality; due to the complex structure, convergence analysis of the

proposed algorithms is not possible and thus unexplored in [29] and [30]. However, it is noted that optimality does not necessarily guarantee convergence.

In this piece of research, we develop an algorithm to simultaneously predict the inputs and states for discrete-time linear systems with direct feedthrough, such as ones in [30]. Though attempting to minimize both mean square errors and error variances, the developed algorithm is only sub-optimal as some optimality is sacrificed to make the algorithm convergent. Compared with [30], the purpose and main contribution of this paper lies in proposing an algorithm with stability properties proven rigorously.

2.2 Problem Formulation

Consider the linear time-invariant dynamic system as shown in Fig. 2.1.

$$\begin{cases} x_{k+1} &= Ax_k + Bu_k + w_k, \\ y_k &= Cx_k + Du_k + v_k, \end{cases} \quad (2.1)$$

where $x_k \in \mathbb{R}^n$ denotes the system state variable at time instant k , $u_k \in \mathbb{R}^m$ is the unknown input, and $y_k \in \mathbb{R}^p$ is the system measurement. A , B , C and D , are known system matrices with compatible dimensions. The process noise w_k and measurement noise v_k are assumed to be mutually uncorrelated zero-mean white noises with known covariances, namely,

$$E\{w_k w_l^T\} = R_w \delta_{k-l}, \quad E\{v_k v_l^T\} = R_v \delta_{k-l}, \quad E\{w_k v_l^T\} = 0,$$

where δ_k is the Kronecker delta function, $R_w > 0$, and $R_v > 0$ are variances of w and v , respectively.

For system (2.1), it is desirable to develop optimal recursive input and state estimators. Here the optimality is defined in the sense of both minimum mean square error (MMSE) and minimum variance (MV). Moreover, the developed SISE algorithm has to be numerically stable and the asymptotical convergence of the estimators has to be guaranteed.

Inspired by the theory of observer design for deterministic linear systems [7], the input and state estimators are designed, respectively, as

$$\hat{u}_k = H_k (y_k - C\hat{x}_k), \quad (2.2)$$

$$\hat{x}_{k+1} = A\hat{x}_k + B\hat{u}_k + L_k (y_k - C\hat{x}_k - D\hat{u}_k), \quad (2.3)$$

where \hat{x}_k represents the state estimate and \hat{u}_k the input estimate. H_k and L_k are estimators' gain matrices that will be designed later. The mean square errors of input and state estimates are defined, respectively, as

$$J_k^u = E \{ \tilde{u}_k^T \tilde{u}_k \}, \quad (2.4)$$

$$J_{k+1}^x = E \{ \tilde{x}_{k+1}^T \tilde{x}_{k+1} \}, \quad (2.5)$$

where \tilde{u}_k and \tilde{x}_{k+1} are estimation errors:

$$\tilde{u}_k = u_k - \hat{u}_k, \quad \tilde{x}_{k+1} = x_{k+1} - \hat{x}_{k+1}.$$

The related covariance matrices of the estimation errors are defined as

$$P_k^u = E \{ \tilde{u}_k \tilde{u}_k^T \}, \quad (2.6)$$

$$P_k^{ux} = E \{ \tilde{u}_k \tilde{x}_k^T \}, \quad (2.7)$$

$$P_{k+1}^x = E \{ \tilde{x}_{k+1} \tilde{x}_{k+1}^T \}. \quad (2.8)$$

It is straightforward to see that matrices P_k^u and P_{k+1}^x are symmetric and positive definite.

Our objectives in the research are to:

- P1. Design the state and input estimators, that is, determining estimator gains L_k and H_k in (2.2) and (2.3);
- P2. Analyze the stability properties of the proposed estimation algorithm.

2.3 Derivation of the Algorithm

This section focuses on the development of the simultaneous input and state estimation algorithm. We start with the optimal design, that is, minimizing both J_k^u and P_k^u to derive L_k , and minimizing both J_{k+1}^x and P_{k+1}^x to obtain H_k . However, it is found that the optimal gain matrices L_k and H_k given by analysis cannot lead to a numerically feasible algorithm as the result of a necessary inversion of a singular matrix. Therefore, we propose some modifications to construct the algorithm.

2.3.1 Preliminaries

To study the optimality property of the proposed SISE algorithm, it is important to ensure that the estimates are unbiased.

Lemma 2.1. For system (2.1), for unbiased input and state estimates in (2.2) and (2.3), D must be of full column rank, and the following initial condition must be satisfied:

$$\hat{x}_0 = E(x_0). \quad (2.9)$$

Proof. Substituting the state equations (2.1) into (2.2) and (2.3), respectively, we obtain

$$\tilde{u}_k = -H_k(C\tilde{x}_k + v_k) + (I - H_kD)u_k, \quad (2.10)$$

$$\tilde{x}_{k+1} = (A - L_kC)\tilde{x}_k + (B - L_kD)\tilde{u}_k - L_kv_k + w_k, \quad (2.11)$$

where I is the identity matrix. Recursively applying the above dynamics until $k = 0$, it can be seen that the estimates are unbiased, namely, $E(\tilde{u}_k) = 0$ and $E(\tilde{x}_k) = 0$, provided both (2.9) and the following input unbiasedness constraint are satisfied

$$H_kD = I. \quad (2.12)$$

From constraint (2.12), D should be of full column rank. Proof of Lemma 2.1 is completed. \square

Under the above unbiasedness condition, (2.10) is then simplified as

$$\tilde{u}_k = -H_k(C\tilde{x}_k + v_k). \quad (2.13)$$

2.3.2 Input Estimation

The optimal input estimator gain matrix, denoted by H_k^* , can be found by solving the following *simultaneous* optimization problem of minimizing J_k^u and P_k^u :

$$H_k^* = \arg \min_{H_k} \{J_k^u, P_k^u\}, \quad \text{s.t. } H_kD = I.$$

In the following, we first show that an H_k^* can minimize J_k^u subject to (2.12) (Theorem 2.1), and that the same H_k^* further minimizes P_k^u under the same constraint (Theorem 2.2).

Theorem 2.1. Assume that input estimation is unbiased. If the optimal gain matrix H_k^* is

$$H_k^* = (D^T Q_k^{-1} D)^{-1} D^T Q_k^{-1}, \quad (2.14)$$

where $Q_k = CP_k^x C^T + R_v$, then the mean square error J_k^u is minimized.

Proof. Using (2.13), J_k^u can be expanded as

$$\begin{aligned}
J_k^u &= \mathbb{E} \left\{ (C\tilde{x}_k + v_k)^T H_k^T H_k (C\tilde{x}_k + v_k) \right\} \\
&= \mathbb{E} \left\{ \tilde{x}_k^T C^T H_k^T H_k C \tilde{x}_k \right\} + \mathbb{E} \left\{ v_k^T H_k^T H_k v_k \right\} \\
&= \text{tr} \left\{ H_k C P_k^x C^T H_k^T \right\} + \text{tr} \left\{ H_k R_v H_k^T \right\} \\
&= \text{tr} \left\{ H_k Q_k H_k^T \right\}.
\end{aligned} \tag{2.15}$$

Let λ be a weighting matrix of appropriate dimension. By using the Lagrange multipliers approach, the above equation can be equivalently written as

$$J_k^u = \text{tr} \left\{ H_k Q_k H_k^T + \lambda (I - H_k D) \right\}. \tag{2.16}$$

Taking the partial derivative of J_k^u with respect to (w.r.t.) H_k to be zero, we have

$$\frac{\partial J_k^u}{\partial H_k} = 2H_k Q_k - \lambda^T D^T = 0. \tag{2.17}$$

Combining (2.17) with (2.12), the optimal H_k is obtained

$$H_k^* = (D^T Q_k^{-1} D)^{-1} D^T Q_k^{-1}. \tag{2.18}$$

This proves Theorem 2.1. \square

Next, a question arises: Can the same H_k^* in (2.14) minimize the error covariance of the input estimation P_k^u under the unbiasedness constraint? Before answering this question in Theorem 2.2, we rewrite P_k^u by incorporating (2.13) as follows:

$$P_k^u = H_k Q_k H_k^T. \tag{2.19}$$

Theorem 2.2. For any H_k satisfying (2.12), the following relation holds true:

$$P_k^u \geq (D^T Q_k^{-1} D)^{-1}, \tag{2.20}$$

where the equality is held if and only if $H_k = H_k^*$.

Proof. Using (2.12), (2.14), and (2.19), we obtain

$$[H_k - H_k^*] Q_k [H_k - H_k^*]^T = H_k Q_k H_k^T - (D^T Q_k^{-1} D)^{-1} = P_k^u - (D^T Q_k^{-1} D)^{-1} \geq 0.$$

Hence (2.20) is proven. The two sides of (2.20) are obviously equal when $H_k = H_k^*$. The uniqueness of H_k^* comes directly from the fact that matrix Q_k is positive definite. \square

With the optimal H_k^* and from (2.7) and (2.13), we have

$$P_k^{ux} = \mathbb{E} \left\{ -H_k^* [C\tilde{x}_k + v_k] \tilde{x}_k^T \right\} = -H_k^* C P_k^x. \tag{2.21}$$

2.3.3 State Estimation

Now let us consider the state estimation problem. Similarly we aim to design the optimal L_k , denoted by L_k^* , by minimizing J_{k+1}^x and P_{k+1}^x *simultaneously*:

$$L_k^* = \arg \min_{L_k} \{J_{k+1}^x, P_{k+1}^x\}.$$

Define the following matrices

$$S_k = MO_k M^T, \quad (2.22)$$

$$T_k = MO_k N^T - BH_k^* R_v, \quad (2.23)$$

$$U_k = NO_k N^T + R_v - DH_k^* R_v - R_v H_k^{*T} D^T, \quad (2.24)$$

where

$$M = \begin{bmatrix} A & B \end{bmatrix}, N = \begin{bmatrix} C & D \end{bmatrix}, O_k = \begin{bmatrix} P_k^x & (P_k^{ux})^T \\ P_k^{ux} & P_k^u \end{bmatrix}. \quad (2.25)$$

Before we proceed to obtain the optimal gain matrix L_k^* , we have the following property for U_k .

Proposition 2.1. For any k , U_k in (2.24) is singular, that is, $\det U_k = 0$.

Proof. Expanding U_k gives

$$U_k = (I - DH_k^*)(CP_k^x C^T + R_v)(I - DH_k^*)^T,$$

from which it follows that

$$\det U_k = \det(I - DH_k^*) \det(CP_k^x C^T + R_v) \det(I - DH_k^*)^T.$$

As $\det(I - DH_k^*) = \det(I - H_k^* D) = 0$, we have $\det U_k = 0$. □

From (2.5) and (2.11), it follows that

$$\begin{aligned} J_{k+1}^x &= \mathbb{E} \{ \tilde{x}_k^T [A - L_k C]^T [A - L_k C] \tilde{x}_k \} + \mathbb{E} \{ \tilde{u}_k^T [B - L_k D]^T [B - L_k D] \tilde{u}_k \} \\ &\quad + 2\mathbb{E} \{ \tilde{x}_k^T [A - L_k C]^T [B - L_k D] \tilde{u}_k \} - 2\mathbb{E} \{ \tilde{u}_k^T [B - L_k D]^T L_k v_k \} \\ &\quad + \mathbb{E} \{ v_k^T L_k^T L_k v_k \} + \mathbb{E} \{ w_k^T w_k \} \\ &= \text{tr} \{ [A - L_k C] P_k^x [A - L_k C]^T \} + \text{tr} \{ [B - L_k D] P_k^u [B - L_k D]^T \} \\ &\quad + 2\text{tr} \{ [B - L_k D] P_k^{ux} [A - L_k C]^T \} - 2\text{tr} \{ L_k v_k \tilde{u}_k^T [B - L_k D]^T \} \\ &\quad + \text{tr} \{ L_k R_v L_k^T \} + \text{tr}(R_w) \\ &= \text{tr} \{ S_k - T_k L_k^T - L_k T_k^T + L_k U_k L_k^T + R_w \}. \end{aligned} \quad (2.26)$$

From (2.8) and (2.11), we have

$$P_{k+1}^x = S_k - L_k T_k^T - T_k L_k^T + L_k U_k L_k^T + R_w. \quad (2.27)$$

Equating the partial derivative of J_{k+1}^x w.r.t L_k to zero, we obtain

$$\frac{\partial J_{k+1}^x}{\partial L_k} = -2T_k + 2L_k U_k = 0. \quad (2.28)$$

Note that matrix U_k is singular, then solution of (2.28) for the gain matrix L_k^* is not unique. Thus to obtain a *numerically stable* algorithm, we seek for an alternative construction of related matrices. Let us define

$$\begin{aligned} \hat{P}_{k+1}^x &= \hat{S}_k - \hat{L}_k \hat{T}_k^T - \hat{T}_k \hat{L}_k^T + \hat{L}_k \hat{U}_k \hat{L}_k^T + R_w, \\ \hat{S}_k &= M \hat{O}_k M^T, \\ \hat{T}_k &= M \hat{O}_k N^T, \\ \hat{U}_k &= N \hat{O}_k N^T + R_v, \\ \hat{H}_k^* &= \left(D^T \hat{Q}_k^{-1} D \right)^{-1} D^T \hat{Q}_k^{-1}, \\ \hat{Q}_k &= C \hat{P}_k^x C^T + R_v, \\ \hat{O}_k &= \begin{bmatrix} \hat{P}_k^x & (\hat{P}_k^{ux})^T \\ \hat{P}_k^{ux} & \hat{P}_k^u \end{bmatrix}, \\ \hat{P}_k^{ux} &= -\hat{H}_k^* C \hat{P}_k^x, \\ \hat{P}_k^u &= \hat{H}_k^* \hat{Q}_k \hat{H}_k^{*T}. \end{aligned}$$

It is noted that if the correlation between \tilde{u}_k and v_k can be ignored, the above matrices will be equivalent to their original counterparts. The sub-optimal matrix gain L_k^* can then be calculated as

$$\hat{L}_k^* = \hat{T}_k \hat{U}_k^{-1}. \quad (2.29)$$

\hat{P}_{k+1}^x can be written equivalently as

$$\hat{P}_{k+1}^x = \hat{S}_k - \hat{T}_k \hat{U}_k^{-1} \hat{T}_k^T + \left(\hat{L}_k - \hat{T}_k \hat{U}_k^{-1} \right) \hat{U}_k \left(\hat{L}_k - \hat{T}_k \hat{U}_k^{-1} \right)^T + R_w.$$

If \hat{L}_k is set to be the aforeproposed \hat{L}_k^* in (2.29), \hat{P}_{k+1}^x will achieve its minimum

$$\hat{P}_{k+1}^x = \hat{S}_k - \hat{T}_k \hat{U}_k^{-1} \hat{T}_k^T + R_w = \hat{S}_k - \hat{L}_k^* \hat{T}_k^T + R_w. \quad (2.30)$$

2.3.4 Algorithm Summary

A summary of the proposed algorithm is given in **Algorithm 1**:

Algorithm 1. The SISE algorithm.

- 1: Initialize: $\hat{x}_0 = E(x_0)$, $\hat{P}_0^x = p_0 I$, where p_0 is a large positive value
- 2: **for** $k = 0$ to $N - 1$ **do**
- 3: $\hat{Q}_k = C\hat{P}_k^x C^T + R_v$
- 4: $\hat{H}_k^* = \left(D^T \hat{Q}_k^{-1} D\right)^{-1} D^T \hat{Q}_k^{-1}$, $\hat{u}_k = \hat{H}_k^* (y_k - C\hat{x}_k)$
- 5: $\hat{P}_k^u = \hat{H}_k^* \hat{Q}_k \hat{H}_k^{*T}$
- 6: **if** $k < N - 1$ **then**
- 7: $\hat{P}_k^{ux} = -\hat{H}_k^* C \hat{P}_k^x$
- 8: $\hat{O}_k = \begin{bmatrix} \hat{P}_k^x & (\hat{P}_k^{ux})^T \\ \hat{P}_k^{ux} & \hat{P}_k^u \end{bmatrix}$
- 9: $\hat{S}_k = [A \ B] \hat{O}_k [A \ B]^T$, $\hat{T}_k = [A \ B] \hat{O}_k [C \ D]^T$, $\hat{U}_k = [C \ D] \hat{O}_k [C \ D]^T + R_v$
- 10: $\hat{L}_k^* = \hat{T}_k \hat{U}_k^{-1}$, $\hat{x}_{k+1} = A\hat{x}_k + B\hat{u}_k + \hat{L}_k^* [y_k - C\hat{x}_k - D\hat{u}_k]$
- 11: $\hat{P}_{k+1}^x = \hat{S}_k - \hat{L}_k^* \hat{T}_k^T + R_w$
- 12: **end if**
- 13: **end for**

Remark 2.1. As aforementioned, our purpose is to develop an algorithm proven stable in theory. Thus during the development of **Algorithm 1**, we make a tradeoff between optimality and stability. Specifically, optimality is sacrificed slightly, but at the same time, the algorithm will gain stability properties that can be proven explicitly. However, we note in Section 3.5 that **Algorithm 1**, though not being fully optimal, still has very satisfactory performance.

Remark 2.2. Consider the case of a linear time-varying discrete-time system:

$$\begin{cases} x_{k+1} &= A_k x_k + B_k u_k + w_k, \\ y_k &= C_k x_k + D_k u_k + v_k, \end{cases} \quad (2.31)$$

where A_k , B_k , C_k and D_k may change with time and the properties of the noises w_k and v_k follow the assumptions in Section 4.2. We can extend conveniently **Algorithm 1** to one such system just by replacing A , B , C and D with their time-varying counterparts A_k , B_k , C_k and D_k . The derivation steps are analogous to the above.

2.4 Convergence Analysis

In this section, we analyze the *stability* of **Algorithm 1** by convergence analysis. The major idea-flow of the proof is sketched first:

- By inspecting **Algorithm 1**, it is found that its numerical stability depends on the convergence of \hat{P}_k^x for all matrix variables are functions of \hat{P}_k^x . If \hat{P}_k^x is convergent, then **Algorithm 1** will be convergent accordingly. Therefore, the convergence analysis of the algorithm is reduced to the convergence property of \hat{P}_k^x or equivalently \hat{P}_{k+1}^x .
- To analyze the stability of **Algorithm 1**, it is necessary to analyze the convergence properties of \hat{P}_{k+1}^x . The expression of \hat{P}_{k+1}^x is to be formulated as a Riccati-like matrix equation, for which the convergence of the solution will be analyzed. Some works have been focusing on the convergence analysis of solutions of the Riccati equation and its variants, for example, see [2; 80; 5; 69]. Finally, we will prove that P_{k+1}^x is upper bounded as k approaches infinity under certain conditions.

According to (2.30), P_{k+1}^x is expressed as

$$\hat{P}_{k+1}^x = M\hat{O}_k M^T - M\hat{O}_k N^T \left(N\hat{O}_k N^T + R_v \right)^{-1} N\hat{O}_k M^T + R_w, \quad (2.32)$$

where \hat{O}_k can be considered as a generalized function of \hat{P}_k^x . From (2.32) we define a generalized algebraic Riccati equation (GARE) as follows:

$$g(X) = MO(X)M^T - MO(X)N^T \left(NO(X)N^T + R_v \right)^{-1} NO(X)M^T + R_w. \quad (2.33)$$

Here $O(X)$ has the same structure as O_k defined by (2.25), with P_k^x replaced by X as follows:

$$O(X) = \begin{bmatrix} X & (P_{uX})^T \\ P_{uX} & P_u \end{bmatrix},$$

where

$$P_{uX} = - \left[D^T(CXC^T + R_v)^{-1}D \right]^{-1} D^T(CXC^T + R_v)^{-1}CX$$

and

$$P_u = \left[D^T(CXC^T + R_v)^{-1}D \right]^{-1}.$$

Similar to the construction of \hat{O}_k for \hat{P}_k^x being positive definite, we assume that X is also positive definite in $O(X)$. By (2.33), (2.32) can be written as an iterative equation

$$\hat{P}_{k+1}^x = g(\hat{P}_k^x). \quad (2.34)$$

Lemma 2.2. $O(X)$ is symmetric, positive definite and monotonically increasing with X .

Proof. The proof is straightforward and thus is omitted. \square

Define a Riccati operator

$$\phi(K, X) = FO(X)F^T + V, \quad (2.35)$$

where $F = M + KN$, $V = KR_vK^T + R_w$. Some properties of the operator are summarized in the following lemma.

Lemma 2.3. The following facts hold true:

- a) With $K_X = -MO(X)N^T(NO(X)N^T + R_v)^{-1}$, $g(X) = \phi(K_X, X)$;
- b) $g(X) = \min_K \phi(K, X)$, $\forall K$;
- c) For $0 < X \leq Y$, $g(X) \leq g(Y)$.

Proof. a) Plugging $K = K_X$ into (2.35) immediately gives fact a).

b) It is straightforward to obtain

$$\phi(K, X) = (M + KN)O(X)(M + KN)^T + KR_vK^T + R_w$$

We can solve for K that minimizes $\phi(K, X)$ by letting

$$\frac{\partial \phi(K, X)}{\partial K} = 0.$$

It is found that $K = K_X$ is the solution of the above equation.

c) As $O(X)$ monotonically increases with X , so does $g(X)$. \square

Lemma 2.4. [69] Assume that $h(\cdot)$ is a monotonically increasing function. If $X_{k+1} = h(X_k)$ and $Y_{k+1} = h(Y_k)$, then

$$X_1 \geq X_0 \implies X_{k+1} \geq X_k,$$

$$X_1 \leq X_0 \implies X_{k+1} \leq X_k,$$

$$X_0 \leq Y_0 \implies X_k \leq Y_k.$$

To proceed further, let us define a new operator:

$$\psi(X) = FO(X)F^T.$$

Variable K in $\psi(X)$ is dropped here since we only need to study the effects of X . It is straightforward to observe that $\psi(X)$ is linear, positive definite, and monotonically increasing. Moreover, $\phi(K, X) = \psi(X) + V$.

Lemma 2.5. Assume there exists $0 < \tilde{X} < \infty$ such that $\tilde{X} > \psi(\tilde{X})$. Consider $X_{k+1} \leq \psi(X_k) + \Delta$ with $\Delta \geq 0$ and initial value $X_0 \geq 0$. Then the sequence X_k is upper bounded.

Proof. For any X , there exist $m \geq 0$ and $0 \leq r < 1$ such that $X \leq m\tilde{X}$ and $\psi(\tilde{X}) < r\tilde{X}$. Then it follows from Lemma 2.4 that

$$0 \leq \psi^k(X) \leq mr^{k-1}\psi(\tilde{X}) \leq mr^k\tilde{X},$$

where

$$\psi^k(X) = \underbrace{\psi(\cdots \psi(\psi(X)) \cdots)}_k \text{ and } \psi^0(X) = X.$$

Thus as $k \rightarrow \infty$, $mr^k\tilde{X} \rightarrow 0$, and further, $\psi^k(X) \rightarrow 0$. Using this conclusion, we have

$$X_k \leq \psi^k(X_0) + \sum_{i=0}^{k-1} \psi^i(\Delta) \leq \left(m_{X_0} r^k + \sum_{i=0}^{k-1} m_{\Delta} r^i \right) \tilde{X} \leq \left(m_{X_0} + \frac{m_{\Delta}}{1-r} \right) \tilde{X},$$

where $m_{X_0} \geq 0$ and $m_{\Delta} \geq 0$. Thus the lemma is proven. \square

Finally, the following theorem establishes the convergence property of Algorithm 1.

Theorem 2.3. Assume that there exists a \tilde{K} and a $\tilde{P} > 0$ such that

$$\tilde{P} > \phi(\tilde{K}, \tilde{P}).$$

Then, for any $\hat{P}_0 \geq 0$, the sequence $\{\hat{P}_k^x\}$ that is generated by the iterative equation $\hat{P}_{k+1}^x = g(\hat{P}_k^x)$ converges, namely,

$$\lim_{k \rightarrow \infty} \hat{P}_k^x = \overline{\tilde{P}},$$

where $\overline{\tilde{P}}$ satisfies $\overline{\tilde{P}} = g(\overline{\tilde{P}})$.

Proof. Note that

$$\tilde{P} > \phi(\tilde{K}, \tilde{P}) = \psi(\tilde{P}) + \tilde{K}R_v\tilde{K}^T + R_w \geq \psi(\tilde{P})$$

Then from Lemma 2.3 we get

$$\hat{P}_{k+1}^x = g(P_k) \leq \phi(\tilde{K}, \hat{P}_k^x) = \psi(\hat{P}_k^x) + \tilde{K}R_v\tilde{K}^T + R_w.$$

Letting $\Delta = \tilde{K}R_v\tilde{K}^T + R_w$ and using Lemma 2.5, we assert that $\{\hat{P}_k^x\}$ is upper bounded.

Next, we shall show the GARE converges to \bar{P} from any $\hat{P}_0 \geq 0$. First, consider the extreme case when $\hat{P}_0 = 0$. Then,

$$0 = \hat{P}_0 \leq g(\hat{P}_0) = \hat{P}_1,$$

which implies by Lemma 2.3-c) and Lemma 2.4 that

$$\hat{P}_0 \leq \hat{P}_1 \leq \hat{P}_2 \leq \dots$$

Yet this sequence is still bounded, as aforeproved. Its upper bound, \bar{P} , is given by the solution of

$$\bar{P} = g(\bar{P}).$$

Now consider the case when $\hat{P}_0 > \bar{P}$. Define

$$K_{\bar{P}} = -A\bar{P}X \left(C\bar{P}C^T + R_v \right)^{-1}, \quad F_{\bar{P}} = A + K_{\bar{P}}C, \quad \psi_{\bar{P}}(X) = F_{\bar{P}}^T X F_{\bar{P}}.$$

We can see that

$$\bar{P} = g(\bar{P}) = \phi(K_{\bar{P}}, \bar{P}) > \psi_{\bar{P}}(\bar{P}).$$

Thus from Lemma 2.5, we get

$$\lim_{k \rightarrow \infty} \psi_{\bar{P}}^k(X) = 0 \text{ for any } X > 0. \quad (2.36)$$

Since $\hat{P}_0 > \bar{P}$, it holds that

$$\hat{P}_1 = g(\hat{P}_0) \geq g(\bar{P}) = \bar{P},$$

which shows

$$\hat{P}_k^x \geq \bar{P}.$$

Let us consider the sequence $\{\hat{P}_k^x - \bar{P}\}$. We have

$$\begin{aligned} 0 &\leq (\hat{P}_{k+1}^x - \bar{P}) = g(\hat{P}_k^x) - g(\bar{P}) = \phi(K_{\hat{P}_k^x}, \hat{P}_k^x) - \phi(K_{\bar{P}}, \bar{P}) \\ &\leq \phi(K_{\bar{P}}, \hat{P}_k^x) - \phi(K_{\bar{P}}, \bar{P}) = \psi_{\bar{P}}(\hat{P}_k^x - \bar{P}). \end{aligned}$$

This, together with (2.36), implies that

$$\lim_{k \rightarrow \infty} \hat{P}_{k+1}^x = \bar{P}.$$

Last, if $0 \leq \hat{P}_0 \leq \bar{P}$, then $g^k(0) < \hat{P}_{k+1}^x = g^k(\hat{P}_0) < g(\bar{P}) = \bar{P}$ and, thus, if $k \rightarrow \infty$, we have $\lim_{k \rightarrow \infty} \hat{P}_{k+1}^x = \bar{P}$, which shows that the sequence $\hat{P}_{k+1}^x = g(\hat{P}_k^x)$ is monotonically convergent to \bar{P} , $\forall \hat{P}_0 \geq 0$. \square

Remark 2.3. The stability of **Algorithm 1** can be quickly concluded from Theorem 2.3, which presents a sufficient condition that guarantees the convergence of \hat{P}_{k+1}^x . Recall that it is

$$\exists \tilde{K} \text{ and } \tilde{P} > 0, \tilde{P} > \phi(\tilde{K}, \tilde{P}).$$

If the condition is satisfied, then \hat{P}_{k+1}^x converges to solution of the GARE, regardless of the initial \hat{P}_0 .

Suppose $\hat{P}_{k+1}^x \rightarrow \bar{P}$ as $k \rightarrow \infty$ by Theorem 2.3. As a consequence, \hat{L}_k^* and \hat{H}_k^* converge to fixed values, i.e., \bar{L} and \bar{H} , respectively. The following theorem can be established.

Theorem 2.4. Assume there exists $0 < \tilde{X} < \infty$ such that $\tilde{X} > \varphi(\tilde{X})$, where $\varphi(\tilde{X}) = (\bar{L}N - M)O(\tilde{X})(\bar{L}N - M)^T$. Then, for any initial value $P_0 \geq 0$, the sequence $\{P_k^x\}$ is upper bounded as $k \rightarrow \infty$.

Proof. From (2.27) it follows that

$$P_{k+1}^x = S_k - \hat{L}_k^* T_k^T - T_k \hat{L}_k^{*T} + \hat{L}_k^* U_k \hat{L}_k^{*T} + R_w.$$

As $k \rightarrow \infty$, we have

$$\begin{aligned} \lim_{k \rightarrow \infty} P_{k+1}^x &= \lim_{k \rightarrow \infty} \left[MO_k M^T - \bar{L}(MO_k N^T - B\bar{H}R_v)^T - (MO_k N^T - B\bar{H}R_v)\bar{L}^T \right. \\ &\quad \left. + \bar{L}(NO_k N^T + R_v - D\bar{H}R_v - R_v(D\bar{H})^T)\bar{L}^T + R_w \right] \\ &= \lim_{k \rightarrow \infty} \left[(\bar{L}N - M)O_k(\bar{L}N - M)^T + \Lambda \right], \end{aligned}$$

where

$$\Lambda = \bar{L}(B\bar{H}R_v)^T + (B\bar{H}R_v)\bar{L}^T + \bar{L} \left[R_v - D\bar{H}R_v - R_v(D\bar{H})^T \right] \bar{L}^T + R_w.$$

There always exists a $\Xi \geq 0$ such that $\Lambda \leq \Xi$. Using Lemme 2.5, we conclude that $\lim_{k \rightarrow \infty} P_{k+1}^x$ is upper bounded. \square

Table 2.1: Example 1: Comparisons between the norms of input and state estimation errors by Algorithm 1 and the Gillijns-Moor algorithm.

		Algorithm 1	Gillijns-Moor algorithm
Input estimation	$\ \tilde{u}_k\ _2$	17.6932	17.0132
	$\ \tilde{u}_k\ _\infty$	4.7194	3.6580
State estimation	$\ \tilde{x}_{1k}\ _2$	11.4800	11.5034
	$\ \tilde{x}_{1k}\ _\infty$	3.2039	3.1733
	$\ \tilde{x}_{2k}\ _2$	6.9794	6.9855
	$\ \tilde{x}_{2k}\ _\infty$	1.4978	1.5098

Remark 2.4. Based on Theorem 2.3, Theorem 2.4 shows that under certain condition the true error variance has an upper bound. In simulation, it is observed that P_{k+1}^x is also convergent to a fixed value. However, to analyze the exact convergence properties of P_{k+1}^x is a very challenging problem, which requires further research.

2.5 Numerical Examples

In this section, two numerical examples are given to illustrate the effectiveness of **Algorithm 1**.

Example 1: Consider an LTI system described by

$$A = \begin{bmatrix} 0.67 & 0 \\ 0 & 0.53 \end{bmatrix}, \quad B = \begin{bmatrix} 1.00 \\ 0.53 \end{bmatrix}, \quad C = \begin{bmatrix} 0.55 & 0.11 \\ 0 & 0.29 \end{bmatrix}, \quad D = \begin{bmatrix} 0.40 \\ 0.20 \end{bmatrix},$$

with

$$R_w = R_v = \begin{bmatrix} 0.08 & 0 \\ 0 & 0.08 \end{bmatrix}.$$

In this example, the input $\{u_k\}$ is taken as a uniformly-distributed sequence with the following properties:

$$E(u_k) = 0, \quad E(u_k^2) = 10, \quad E(u_k u_l) = 0 \text{ for } k \neq l.$$

Suppose that no information about $\{u_k\}$ is available for use. In the simulation, only the

output $\{y_k\}$ is present and **Algorithm 1** is applied. The results observed are summarized as follows:

1. The input estimation and state estimation results are shown in Fig. 2.2, respectively. It is seen that input estimates are close to the actual inputs. Meanwhile, estimation of the two states also exhibits good performance as there are only trivial differences between the estimates and their true values.
2. For initialization, set $\hat{P}_0^x = 10^6 I$. In the simulation, we find that \hat{P}_k^x quickly converges from \hat{P}_0^x to \bar{P} :

$$\bar{P} = \begin{bmatrix} 0.5037 & 0.2118 \\ 0.2118 & 0.2256 \end{bmatrix}.$$

It is straightforward to check that $\bar{P} = g(\bar{P})$ holds true, which indicates \bar{P}^x to be the stable solution to the GARE. This result confirms the convergence analysis in Section 2.4. Meanwhile, we note that the true error variance, P_k^x also converges to a fix point \bar{P} :

$$\bar{P} = \begin{bmatrix} 0.8671 & 0.4185 \\ 0.4185 & 0.3431 \end{bmatrix}.$$

The trace of P_k^x , shown in Fig. 2.3 (blue solid line), demonstrates the monotonically converging trend. This suggests that, in addition to being upper bounded, P_k^x is very likely to have certain convergence property. Yet to prove this requires further research.

3. Comparisons are made between the algorithm in [30] (referred as the Gillijns-Moor algorithm) and Algorithm 1. First, in Fig. 2.3, it is shown that $\text{tr}(P_k^x)$ yielded by the Gillijns-Moor algorithm (red dashed line) is slightly smaller than that by **Algorithm 1**. Second, the consequent input estimation errors \tilde{u}_k and the state estimation errors \tilde{x}_k are compared between both algorithms. The comparison results of errors' 2-norm and ∞ -norm values in one implementation are shown in Table 1. From Fig. 2.3 and Table 1, we note that both algorithms have very close performance.

Example 2: In this example, we study the algorithm performance under strong noises. Now suppose

$$R_w = R_v = \begin{bmatrix} 0.8 & 0 \\ 0 & 0.8 \end{bmatrix}.$$

Table 2.2: Example 2: Comparisons between the norms of input and state estimation errors by Algorithm 1 and the Gillijns-Moor algorithm.

		Algorithm 1	Gillijns-Moor algorithm
Input estimation	$\ \tilde{u}_k\ _2$	46.5408	45.8290
	$\ \tilde{u}_k\ _\infty$	10.2476	10.3013
State estimation	$\ \tilde{x}_{1k}\ _2$	31.4444	31.5030
	$\ \tilde{x}_{1k}\ _\infty$	6.6983	6.7860
	$\ \tilde{x}_{2k}\ _2$	19.5499	19.6002
	$\ \tilde{x}_{2k}\ _\infty$	4.4089	4.3822

The estimation results by **Algorithm 1** are given in Fig. 2.6; the trace and norm comparison results between **Algorithm 1** and the Gillijns-Moor algorithm are shown in Fig. 2.5 and Table 2.

In fact, through numerous simulations for the purpose of comparison (simulation results are not included here because of space limit, but available upon request), we consistently observe that the performances of both **Algorithm 1** proposed and the Gillijns-Moor algorithm are comparable. On the other hand, an obvious advantage when applying **Algorithm 1** is that the convergence can be verified in advance using the analysis provided in Section 2.4.

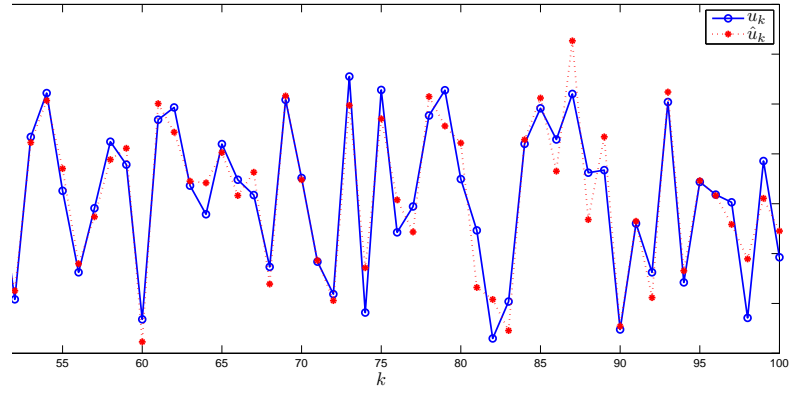
In some applications such as maneuvering target tracking, certain *a priori* information can be assumed about the input [51], and then it is likely to help improve estimation performance.

Example 3: Consider again the system in Example 1. However, the input $\{u_k\}$ is chosen to be a random binary-value signal taking either -10 or 10. Assume that the possible binary values are known to us. It is shown in Fig. 2.6a that the inputs and their estimates are accurately superimposed. The state estimation, as illustrated in Fig. 2.6, is also precise; further, the state estimation performance is better than that in Example 1, since part of the input signal's information is known *a priori*. Both examples have the same GARE, therefore the convergence pattern of $\text{tr}(P_k^x)$ in this example exhibits a very similar trend as the one in Example 1 (omitted here), which verifies the convergence of the input and

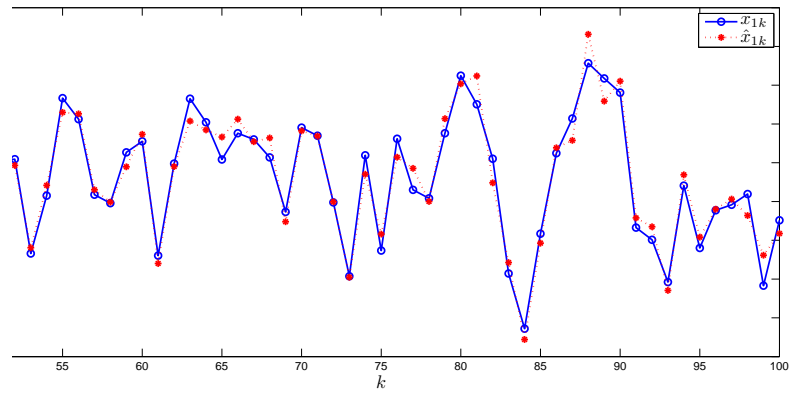
state estimation.

2.6 Summary

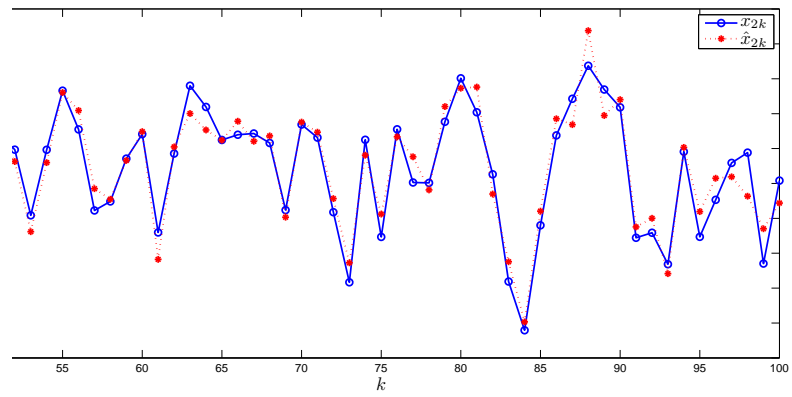
We have studied the problem of simultaneous input and state estimation for systems with direct feedthrough. The challenge of the problem lies in limited data information. Starting from the optimality (MMSE and MV) analysis, optimal design procedures for the SISE problem are presented. To ensure the stability of the algorithm, we modify the procedures slightly to develop the input and state co-estimation algorithm. Further, the stability of the proposed algorithm is analyzed in detail, showing that the estimation error variance is upper bounded. Several numerical examples are presented to demonstrate the effectiveness of the proposed algorithm and compared with existing results.



(a)



(b)



(c)

Figure 2.2: Example 1: Results for input and state estimation. (a) The input u_k and its estimate \hat{u}_k . (b) The first state x_{1k} and its estimate \hat{x}_{1k} . (c) The second state x_{2k} and its estimate \hat{x}_{2k} .

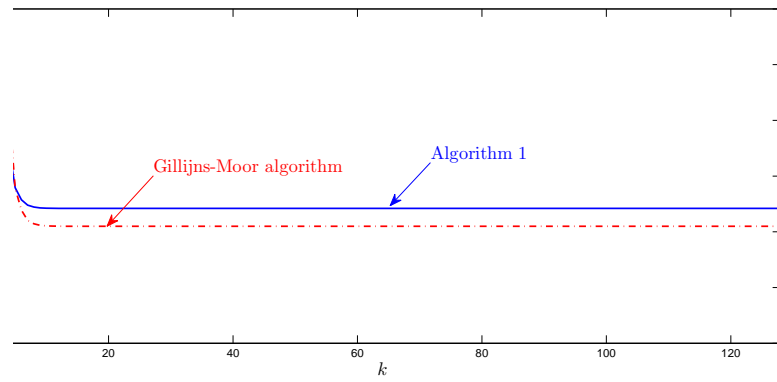
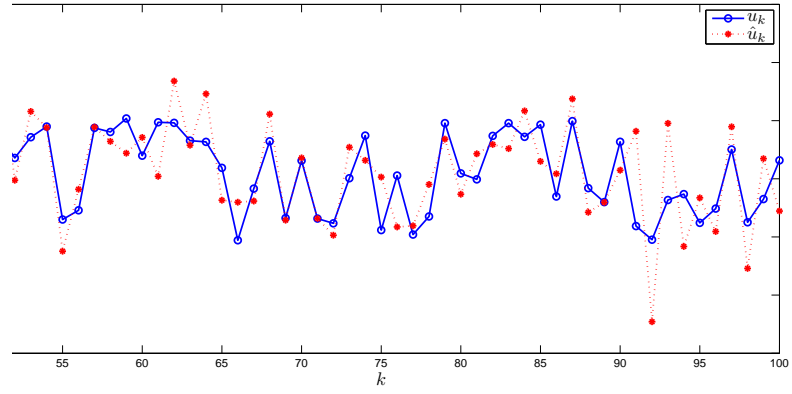
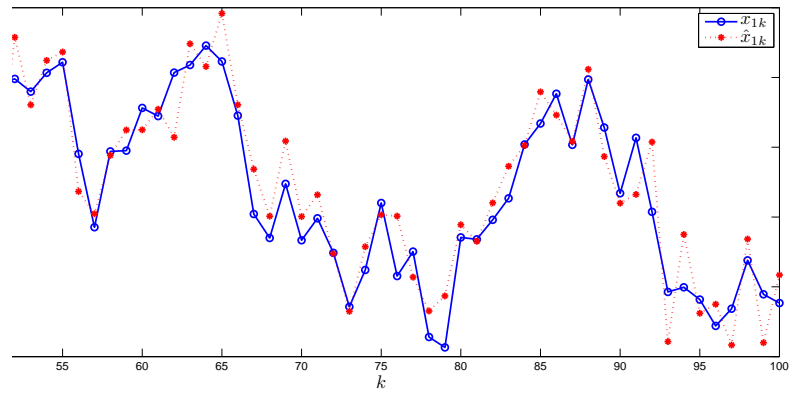


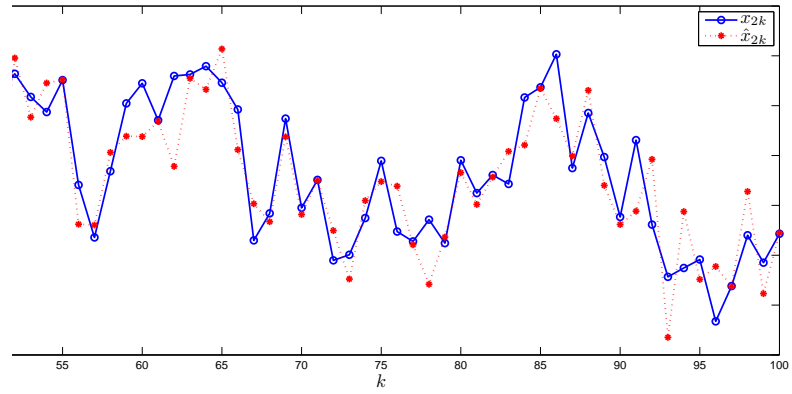
Figure 2.3: Example 1: Trace of P_k^x vs. k .



(a)



(b)



(c)

Figure 2.4: Example 2: Results for input and state estimation. (a) The input u_k and its estimate \hat{u}_k . (b) The first estimate x_{1k} and its estimate \hat{x}_{1k} . (c) The second x_{2k} and its estimates \hat{x}_{2k} .

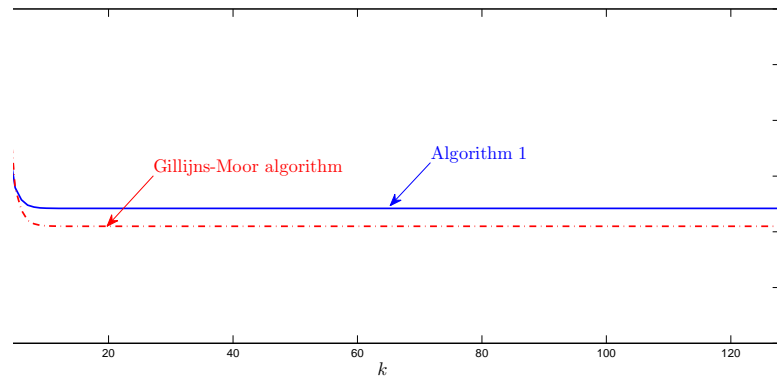
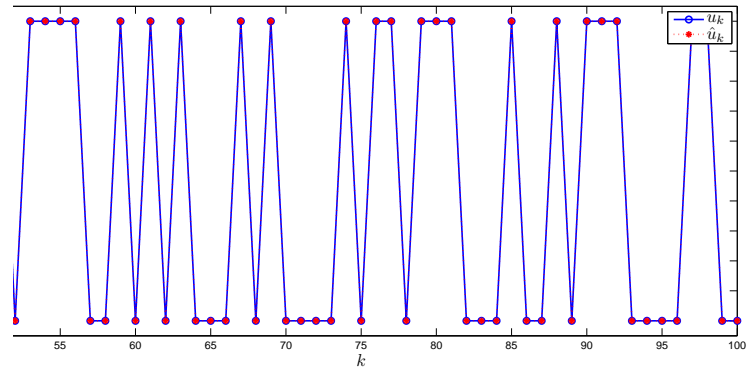
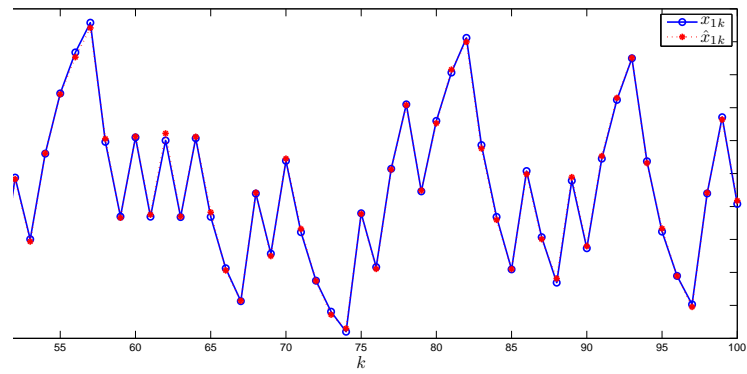


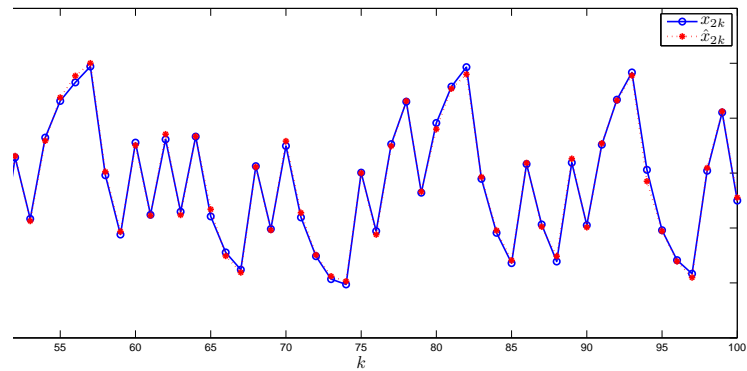
Figure 2.5: Example 2: Trace of P_k^x vs. k .



(a)



(b)



(c)

Figure 2.6: Example 3: Results for input and state estimation. (a) The input u_k and its estimate \hat{u}_k . (b) The first state x_{1k} and its estimate \hat{x}_{1k} . (c) The second state x_{2k} and its estimate \hat{x}_{2k} .

CHAPTER 3

SYSTEM IDENTIFICATION FOR NETWORKED SYSTEMS

3.1 Introduction and Literature Review

In NCSs, because network-induced time delays and data packet losses worsen the control performance and even destroy the stability, many studies have been devoted to controller design for NCSs, e.g., [11; 73; 34] and the references therein. However, it is known that many control design methodologies depend on dynamic models, so prior to the development of controllers, system identification, aimed at building models from measured data, must be carried out. In this chapter, we thus discuss how to identify model parameters of a plant subject to randomly missing measurements in a network environment, as illustrated in Fig. 3.1.

In an NCS, the plant plus the actuator and sensor are installed at a remote location. At the near end, an input transmitter sends the input signal to excite the plant, and an output receiver collects the plant output. Both are done through a network. A parameter estimation module identifies the parameters of the plant in an online manner. The network is assumed to operate under *TCP-like protocols* which can guarantee an acknowledgement of received packets; it has been widely used in research on state estimation and control over networks [66]. For such a networked system, both input and output are subject to randomly missing due to the nature of communication networks. Obviously, identification over networks is more challenging, and classical parameter estimation methods, such as least squares (LS) [41] and stochastic gradients (SG), can no longer be applied *directly*.

Generally, existing research on identification for systems with incomplete input-output data can be divided into two categories.

- **Systems with regular missing outputs.** Systems with regular missing data can also be viewed as multirate systems which have uniform but various input/output sampling rates [10]. Such systems may have regular-output-missing feature. In [17],

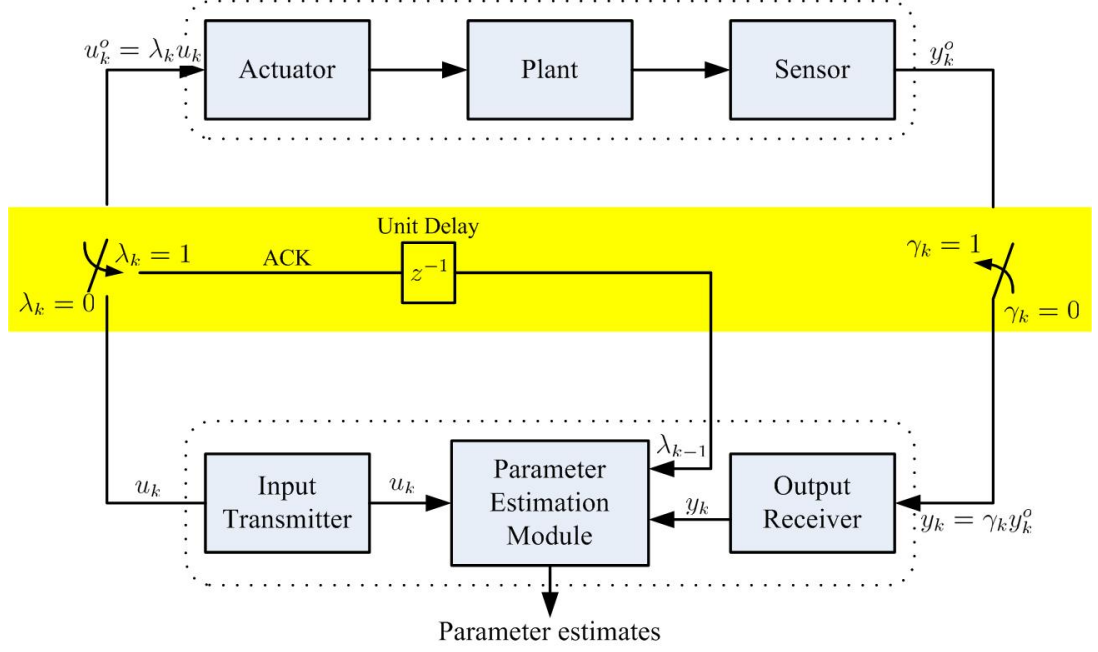


Figure 3.1: Scenario of identification over a lossy network under TCP-like protocols.

an auxiliary model based method is proposed and a modified recursive least squares (RLS) algorithm is developed to simultaneously estimate the system parameter and the unavailable outputs of a dual-rate system. Other related research on identification of multirate system models can be referred to [79; 18; 20; 52], to name a few.

- **Systems with randomly missing outputs.** In many industrial applications non-uniform sampling is widely adopted, i.e., the output is unequally measured at time-varying sampling rates. In [50], LS based estimation of continuous-time autoregressive (AR) models from discrete-time data distributed non-uniformly is studied. Parameter estimation of discrete-time AR models with missing observations is discussed by using a SG-type method [56]. Yet in both works no or only qualitative convergence analysis was given. An LS based scheme is presented in [75] for autoregressive exogenous (ARX) model identification from incomplete input/output data; however, this scheme is not recursive for online use, and it is sensitive to data missing pattern. Albertos *et al.* and Wallin *et al.* study the estimation of missing output measurements [1; 76], which pave the way for further estimating parameters; a systematic description of joint output and parameter estimation with irregular output missing is provided in [65]. Isaksson uses the Kalman filtering and fixed-interval

smoothing techniques to reconstruct the missing data first and then identified ARX models by LS, ML and expectation-maximization (EM) algorithms [45]; the identification process [45], however, is one-step but not recursive. A frequency domain solution is proposed in [64] and it treats all missing measurements as parameters, potentially leading to a large amount of parameters to be identified. An l_p norm parameter estimation algorithm is proposed in [9] to address parameter estimation from input/output data that has missing values as well as noisy disturbances; further, it is nicely shown that: If the probability of the missing measurements is less than $1/2$, then the parameter estimate will converge to the true parameter; however, this algorithm is conservative in two-fold: (1) there is a restrictive assumption that both input and output missing have the same probability, which is not always the case in practical NCSs; (2) the probability of missing data is less than $1/2$.

Parameter estimation subject to data missing is also an important research topic in time series analysis. ML and EM algorithms appear to be popular solutions, by maximizing the likelihood functions of linear or nonlinear models, see [44; 6; 74; 47] and the references therein.

However, we maintain that system identification in a lossy network environment has not been fully investigated, which is the focus of our research. The celebrated Kalman filter has been elegantly applied to estimate system parameters [32; 4]; however, when the networked system is subject to randomly missing measurements, it cannot be directly applied here. The main objectives of this work are three-fold:

- Under the TCP-like protocols, to model the input and output missing as two separate Bernoulli processes with different probabilities of missing data, to design a missing output estimator, and further to develop a modified Kalman filter based recursive algorithm.
- To investigate the performance of the proposed estimation algorithms using the stochastic process theory.
- Furthermore, to establish the performance properties of the missing output estimation.

3.2 Problem Formulation

The identification problem in a TCP-based network environment is shown in Fig. 3.1. Let us consider the following output-error model with intermittent input and output information:

$$x_k^o = \frac{B_z}{A_z} u_k^o, \quad (3.1)$$

$$y_k^o = x_k^o + v_k, \quad (3.2)$$

$$u_k^o = \lambda_k u_k, \quad (3.3)$$

$$y_k = \gamma_k y_k^o, \quad (3.4)$$

where u_k^o is the input to the actuator, u_k is the desired input from the input transmitter, y_k^o is the sensor output, and y_k is output transmitted to the output receiver. A_z and B_z are polynomials in the unit delay operator z^{-1} :

$$A_z = 1 + a_1 z^{-1} + a_2 z^{-2} + \cdots + a_{n_a} z^{-n_a},$$

$$B_z = b_0 + b_1 z^{-1} + b_2 z^{-2} + \cdots + b_{n_b} z^{-n_b}.$$

The polynomial orders n_a and n_b are assumed to be known. Eqns. (3.1)-(3.2) can be written equivalently as the following linear regression model:

$$x_k^o = \varphi_k^{oT} \theta, \quad y_k^o = x_k^o + v_k, \quad (3.5)$$

where

$$\varphi_k^o = [-x_{k-1}^o \ -x_{k-2}^o \ \cdots \ -x_{k-n_a}^o \ u_k^o \ u_{k-1}^o \ \cdots \ u_{k-n_b}^o]^T,$$

$$\theta = [a_1 \ a_2 \ \cdots \ a_{n_a} \ b_0 \ b_1 \ \cdots \ b_{n_b}]^T.$$

Regression vector φ_k^o contains lagged input and output values of the plant, and the parameter vector θ contains model parameters to be estimated.

As aforementioned, the network is lossy and thus results in randomly input-output missing, so Bernoulli random variables λ_k and γ_k are introduced to characterize the data missing pattern. The probability distributions of λ_k and γ_k are defined as

$$P(\lambda_k) = \begin{cases} \lambda, & \text{if } \lambda_k = 1, \\ 1 - \lambda, & \text{else if } \lambda_k = 0, \end{cases} \quad \text{and} \quad P(\gamma_k) = \begin{cases} \gamma, & \text{if } \gamma_k = 1, \\ 1 - \gamma, & \text{else if } \gamma_k = 0. \end{cases}$$

Here, the difference between γ_k and λ_k is noteworthy. Under TCP-like protocols, receipt of a packet u_k is acknowledged by a packet message. Yet the acknowledgement will have a unit-time delay. Therefore, at time instant k , γ_k and λ_{k-1} (instead of λ_k) are known. We can define the following information set:

$$\mathcal{T}_k \triangleq \{\Gamma_k Y_k, U_k, \Lambda_{k-1}\}, \quad (3.6)$$

where $\Gamma_k Y_k = (\gamma_k y_k, \gamma_{k-1} y_{k-1}, \dots, \gamma_1 y_1)$, $U_k = (u_k, u_{k-1}, \dots, u_1)$ and $\Lambda_k = (\lambda_k, \lambda_{k-1}, \dots, \lambda_1)$. In fact, \mathcal{T}_k provides all the data information that is available for system identification.

To this end, the objectives are posed in the following:

- P1. Based on the incomplete input/output data information \mathcal{T}_k , how to estimate the system parameters θ ?
- P2. How to evaluate the accuracy of the parameter estimation and the missing output estimation?

In what follows, a recursive algorithm for identification in a TCP-based network environment will be proposed. The algorithm is developed based on the Kalman filter. Further, convergence properties of the proposed algorithm will be analyzed.

3.3 Derivation of the Algorithm

Consider the model in (3.5). It is shown by [32; 4] that the Kalman filter can be used to perform parameter estimation. By using the auxiliary model, we have the following Kalman filter based algorithm:

$$\hat{\theta}_k = \hat{\theta}_{k-1} + K_k^a (y_k^o - \varphi_k^{aT} \hat{\theta}_{k-1}), \quad (3.7)$$

$$K_k^a = \frac{P_{k-1}^a \varphi_k^a}{r_v + \varphi_k^{aT} P_{k-1}^a \varphi_k^a}, \quad (3.8)$$

$$P_k^a = P_{k-1}^a - \frac{P_{k-1}^a \varphi_k^a \varphi_k^{aT} P_{k-1}^a}{r_v + \varphi_k^{aT} P_{k-1}^a \varphi_k^a}, \quad (3.9)$$

$$x_k^a = \varphi_k^{aT} \hat{\theta}_k, \quad (3.10)$$

$$\varphi_k^a = [-x_{k-1}^a \quad -x_{k-2}^a \quad \dots \quad -x_{k-n_a}^a \quad u_k^o \quad u_{k-1}^o \quad \dots \quad u_{k-n_b}^o]^T, \quad (3.11)$$

where $\hat{\theta}_k$ represents the estimated parameter vector at instant k .

However, due to missing input-output data (as shown in (3.3)-(3.4)), the information available for identification is constrained within \mathcal{T}_k . Hence, the above algorithm will fail if

directly used to estimate the parameters of the networked plant. Thus we are motivated to design a new algorithm accounting for the problem of randomly missing data in a network environment.

From the definition of \mathcal{T}_k and Fig. 3.1, let us take a closer look at how to deal with the problem of randomly input and output missing: (1) Thanks to TCP acknowledgement mechanism, the input to the actuator $u_k^o = \lambda_k u_k$ is available, but there exists a unit time delay. For the input missing, we propose to delay identification for a unit time, and if it is missed, then the input is regarded as 0. (2) The sensor output y_k^o is subject to random loss when transmitted through network to the output receiver. Therefore, an output estimator is to be designed.

Let us begin with output estimator design. Inspired by the work on output prediction by Albertos, *et al.* [1], the following estimator is to be used:

$$z_k = y_k + (1 - \gamma_k)\hat{y}_k, \quad (3.12)$$

where

$$\hat{y}_k = \varphi_k^T \hat{\theta}_k$$

and φ_k will be defined later in (3.17). The above estimator adaptively reconstructs the missing output data. It has an intuitive yet efficient structure, and its convergence properties will be analyzed in Section 3.4.

Now we are in good position to derive the recursive parameter estimation algorithm. Replacing y_k^o in the algorithm (3.7)-(3.11) by z_k , and incorporating the random variable γ_k lead to

$$\hat{\theta}_{k+1}|\mathcal{T}_{k+1} = \hat{\theta}_k + K_{k+1}(z_k - \varphi_k^T \hat{\theta}_k), \quad (3.13)$$

$$K_{k+1}|\mathcal{T}_{k+1} = \frac{P_k \varphi_k}{r_v + \varphi_k^T P_k \varphi_k}, \quad (3.14)$$

$$P_{k+1}|\mathcal{T}_{k+1} = P_k - \gamma_k \frac{P_k \varphi_k \varphi_k^T P_k}{r_v + \varphi_k^T P_k \varphi_k}, \quad (3.15)$$

$$x_k|\mathcal{T}_{k+1} = \varphi_k^T \hat{\theta}_{k+1}, \quad (3.16)$$

$$\varphi_k|\mathcal{T}_{k+1} = [-x_{k-1} \ -x_{k-2} \ \cdots \ -x_{k-n_a} \ \lambda_k u_k \ \lambda_{k-1} u_{k-1} \ \cdots \ \lambda_{k-n_b} u_{k-n_b}]^T. \quad (3.17)$$

For ease of presentation, we shall drop \mathcal{T}_{k+1} and use $\hat{\theta}_{k+1}$, K_{k+1} , P_{k+1} , x_k and φ_k alone instead.

Remark 3.1. A closer scrutiny of (3.13) shows that it can be decomposed into two equations:

$$\begin{cases} \hat{\theta}_{k+1} = \hat{\theta}_k + K_k(y_k - \varphi_k^T \hat{\theta}_k), & \text{if } \gamma_k = 1; \\ \hat{\theta}_{k+1} = \hat{\theta}_k, & \text{else if } \gamma_k = 0. \end{cases}$$

Similarly, we can decompose (3.15) in a similar way. It is noted that $\hat{\theta}_k$ and P_k are updated only when the output is available, and will remain unchanged otherwise.

3.4 Convergence Analysis

To begin our quest of establishing convergence properties of the estimation algorithms, some necessary preliminaries are presented first. The convergence analysis of the parameter estimation in (3.13)-(3.17) and the output estimation in (3.12) are to be carried out under the stochastic framework inspired by [8], [16] and [17].

3.4.1 Preliminaries

Some basic facts about the positive definite matrices will be used in this section and are summarized in the following lemma.

Lemma 3.1. [35] Let A, B be $n \times n$ positive definite matrices with the relation $A \leq B$, and C be a $n \times m$ matrix. Then

$$\lambda_{\min}(A)I \leq A \leq \lambda_{\max}(A)I, \quad (3.18)$$

$$C^T A C \leq C^T B C, \quad (3.19)$$

where I is the identity matrix. If the eigenvalues of A and B are arranged in the same order, then

$$\lambda_i(A) \leq \lambda_i(B) \quad (3.20)$$

for $i = 1, 2, \dots, n$.

To facilitate the convergence analysis, we equivalently rewrite the proposed algorithm, as shown in the following lemma.

Lemma 3.2. The algorithm (3.13)-(3.15) can be written equivalently as follows:

$$\hat{\theta}_{k+1} = \hat{\theta}_k + r_v^{-1} P_{k+1} \varphi_k (z_k - \varphi_k^T \hat{\theta}_k), \quad (3.21)$$

$$P_{k+1}^{-1} = P_k^{-1} + \gamma_k r_v^{-1} \varphi_k \varphi_k^T. \quad (3.22)$$

Proof. The proof is straightforward and thus is omitted here. \square

Further, define

$$\begin{aligned}(P_{k+1}^o)^{-1} &= p_0 I + r_v^{-1} \varphi_k^o \varphi_k^{oT}, \\ r_k^o &= \text{tr}((P_k^o)^{-1}), \\ r_k &= \text{tr}(P_k^{-1}),\end{aligned}$$

where p_0 is a positive real value large enough. The relations between r_k^o and $|P_k^{o-1}|$, and between r_k and $|P_k^{-1}|$, are established in the following lemma.

Lemma 3.3. The following relations hold:

$$\ln |(P_k^o)^{-1}| = O(\ln r_k^o), \quad \ln \mathbb{E}|P_k^{-1}| = O(\ln \mathbb{E} r_k). \quad (3.23)$$

Proof. It can be proved by following the similar line in [17]. \square

The next lemma shows the convergence of three infinite series that will be useful later.

Lemma 3.4. The following inequalities hold:

$$\sum_{i=1}^t r_v^{-1} \mathbb{E}(\gamma_i \varphi_i^T P_{i+1} \varphi_i) \leq \ln \mathbb{E}|P_{k+1}^{-1}| + n_0 \ln p_0 \text{ almost surely (a.s.)} \quad (3.24)$$

$$\sum_{i=1}^{\infty} r_v^{-1} \frac{\mathbb{E}(\gamma_i \varphi_i^T P_{i+1} \varphi_i)}{(\ln \mathbb{E}|P_{i+1}^{-1}|)^c} < \infty \text{ a.s.}, \quad (3.25)$$

$$\sum_{i=1}^{\infty} r_v^{-1} \frac{\mathbb{E}(\gamma_i \varphi_i^T P_{i+1} \varphi_i)}{\ln \mathbb{E}|P_{i+1}^{-1}| (\ln \ln \mathbb{E}|P_{i+1}^{-1}|)^c} < \infty \text{ a.s.}, \quad (3.26)$$

where $n_0 = n_a + n_b$ and $c > 1$.

Proof. The proof can be done along the similar way as Lemma 2 in [18] and is omitted here. \square

The following is the well-known martingale convergence theorem that lays the foundation for the upcoming convergence analysis.

Theorem 3.1. [31] Let $\{X_k\}$ be a sequence of nonnegative random variables adapted to an increasing σ -algebras $\{\mathcal{F}_k\}$. If

$$\mathbb{E}(X_{k+1} | \mathcal{F}_k) \leq (1 + \epsilon_k) X_k - \alpha_k + \beta_k, \quad \text{a.s.}$$

where $\alpha_k \geq 0$, $\beta_k \geq 0$ and $\mathbb{E}X_0 < \infty$, $\sum_{i=0}^{\infty} |\epsilon_i| < \infty$, $\sum_{i=0}^{\infty} \beta_i < \infty$ a.s., then X_k converges almost surely to a finite random variable and

$$\lim_{N \rightarrow \infty} \sum_{i=0}^N \alpha_i < \infty, \quad \text{a.s.}$$

3.4.2 Convergence of Parameter Estimation

We will prove that the parameter estimation algorithm (3.13)-(3.17) is convergent by constructing a martingale process satisfying the conditions of Theorem 3.1. This is the essential of the next theorem.

Theorem 3.2. Assume that the driven noise $\{v_k, \mathcal{F}_k\}$ is a martingale difference sequence adapted to a family of increasing σ -algebras $\{\mathcal{F}_k\}$. For the considered system, the following assumptions are made:

$$(A1) \quad \mathbb{E}(v_k | \mathcal{F}_{k-1}) = 0, \quad \text{a.s.},$$

$$(A2) \quad \mathbb{E}(v_k^2 | \mathcal{F}_{k-1}) = r_v < \infty, \quad \text{a.s.},$$

$$(A3) \quad \exists \alpha_0, \beta_0, c_0 \in R^+ \text{ and } k_0 \in N^+, \alpha_0 I \leq \frac{1}{k} \sum_{i=1}^k \varphi_i^o \varphi_i^{oT} \leq \beta_0 t^{c_0} I, \text{ for } k \geq k_0,$$

$$(A4) \quad G_z = \frac{1}{A_z} - \frac{1}{2} \text{ is strictly positive real.}$$

Then the square parameter estimation error, $\|\hat{\theta}_k - \theta\|^2$, produced by the algorithm (3.13)-(3.17), satisfies

$$(C1) \quad \|\hat{\theta}_k - \theta\|^2 = O\left[\frac{(\ln k)^c}{k}\right] \rightarrow 0 \text{ a.s., } c > 1,$$

$$(C2) \quad \|\hat{\theta}_k - \theta\|^2 = O\left[\frac{\ln k (\ln \ln k)^c}{k}\right] \rightarrow 0 \text{ a.s., } c > 1.$$

Proof. The proof is organized into three parts for ease of reading. First of all, the objective of each part is sketched as follows.

- *Part 1* is to show

$$(P1.1) \quad \|\hat{\theta}_k - \theta\|^2 = O\left[\frac{(\ln \mathbb{E}r_k)^c}{\lambda_{\min}(P_k^{-1})}\right] \text{ a.s., } c > 1,$$

$$(P1.2) \quad \|\hat{\theta}_k - \theta\|^2 = O\left[\frac{\ln \mathbb{E}r_k (\ln \ln \mathbb{E}r_k)^c}{\lambda_{\min}(P_k^{-1})}\right] \text{ a.s., } c > 1.$$

- *Part 2* is to show

$$(P2.1) \quad \|\hat{\theta}_k - \theta\|^2 = O \left[\frac{(\ln r_{0t})^c}{\lambda_{\min}(P_{0t}^{-1})} \right] \text{ a.s., } c > 1,$$

$$(P2.2) \quad \|\hat{\theta}_k - \theta\|^2 = O \left[\frac{\ln r_{0t} (\ln \ln r_{0t})^c}{\lambda_{\min}(P_{0t}^{-1})} \right] \text{ a.s., } c > 1.$$

- Finally *Part 3* will show the conclusions (C1) and (C2).

Part 1:

Define the parameter estimation error vector, innovation vector and residual vector, respectively, as follows:

$$\begin{aligned} \tilde{\theta}_{k+1} &= \hat{\theta}_{k+1} - \theta, \\ e_k &= z_k - \varphi_k^T \hat{\theta}_k, \\ \eta_k &= \gamma_k(y_k^o - x_k). \end{aligned}$$

It can be verified that

$$\eta_k = \gamma_k(x_k^o - x_k + v_k), \quad (3.27)$$

$$\eta_k = (1 - r_v^{-1} \varphi_k^T P_{k+1} \varphi_k) e_k \quad (3.28)$$

$$= (1 + r_v^{-1} \varphi_k^T P_k \varphi_k)^{-1} e_k. \quad (3.29)$$

Further, define the parameter estimation error vector and a Lyapunov-like function as

$$\begin{aligned} \tilde{\theta}_k &= \hat{\theta}_k - \theta, \\ V_k &= \tilde{\theta}_k^T P_k^{-1} \tilde{\theta}_k. \end{aligned}$$

From (3.21) and (3.28), it follows that

$$\begin{aligned} \tilde{\theta}_{k+1} &= \tilde{\theta}_k + r_v^{-1} P_{k+1} \varphi_k e_k \\ &= \tilde{\theta}_k + r_v^{-1} P_k \varphi_k \eta_k. \end{aligned} \quad (3.30)$$

V_{k+1} can be evaluated as

$$\begin{aligned} V_{k+1} &= V_k + \gamma_k r_v^{-1} (\varphi_k^T \tilde{\theta}_k)^2 + 2\gamma_k r_v^{-1} \varphi_k^T \tilde{\theta}_k e_k + \gamma_k r_v^{-2} \varphi_k^T P_{k+1} \varphi_k e_k^2 \\ &= V_k + \gamma_k r_v^{-1} (\varphi_k^T \tilde{\theta}_{k+1})^2 + 2\gamma_k \varphi_k^T \tilde{\theta}_{k+1} (r_v + \varphi_k^T P_k \varphi_k)^{-1} e_k + \\ &\quad \gamma_k r_v^{-2} \varphi_k^T P_{k+1} \varphi_k (r_v^{-1} \varphi_k^T P_{k+1} \varphi_k - 1) e_k^2. \end{aligned}$$

Since $\varphi_k^T P_{k+1} \varphi_k \geq 0$ and $1 - r_v^{-1} \varphi_k^T P_{k+1} \varphi_k \geq 0$, then

$$\begin{aligned} V_{k+1} &\leq V_k + \gamma_k r_v^{-1} (\varphi_k^T \tilde{\theta}_{k+1})^2 + 2\gamma_k \varphi_k^T \tilde{\theta}_{k+1} (r_v + \varphi_k^T P_k \varphi_k)^{-1} e_k \\ &= V_k + \gamma_k r_v^{-1} (\varphi_k^T \tilde{\theta}_{k+1})^2 + 2\gamma_k r_v^{-1} \varphi_k^T \tilde{\theta}_{k+1} (1 - r_v^{-1} \varphi_k^T P_{k+1} \varphi_k) e_k. \end{aligned}$$

Define

$$\begin{aligned} \tilde{u}_k &= -\varphi_k^T \tilde{\theta}_{k+1}, \\ \tilde{y}_k &= \frac{1}{2} \varphi_k^T \tilde{\theta}_{k+1} + y_k^o - \varphi_k^T \hat{\theta}_{k+1} - v_k. \end{aligned}$$

Then it follows that

$$\begin{aligned} V_{k+1} &\leq V_k - 2\gamma_k r_v^{-1} \tilde{u}_k \tilde{y}_k + 2\gamma_k r_v^{-1} \varphi_k^T \tilde{\theta}_k v_k + 2\gamma_k r_v^{-2} \varphi_k^T P_{k+1} \varphi_k e_k v_k \\ &= V_k - 2\gamma_k r_v^{-1} \tilde{u}_k \tilde{y}_k + 2\gamma_k r_v^{-1} \varphi_k^T \tilde{\theta}_k v_k + 2\gamma_k r_v^{-2} \varphi_k^T P_{k+1} \varphi_k [(e_k - \gamma_k v_k) v_k + \gamma_k v_k^2]. \end{aligned} \quad (3.31)$$

Taking expectation with respect to \mathcal{F}_k on both sides of (3.31) gives

$$\mathbb{E}(V_{k+1} | \mathcal{F}_k) \leq V_k - 2\mathbb{E}(\gamma_k \tilde{u}_k \tilde{y}_k | \mathcal{F}_{k-1}) + 2r_v^{-1} \mathbb{E}(\gamma_k \varphi_k^T P_{k+1} \varphi_k). \quad (3.32)$$

We also have

$$\begin{aligned} A_z(y_k^o - \varphi_k^T \hat{\theta}_{k+1} - v_k) &= B_z \lambda_k u_k - A_z x_k \\ &= -x_k + \varphi_k^T \theta \\ &= -\varphi_k^T \tilde{\theta}_{k+1} = \tilde{u}_k, \end{aligned} \quad (3.33)$$

which leads to

$$\tilde{y}_k = \left(\frac{1}{A_z} - \frac{1}{2} \right) \tilde{u}_k.$$

In (A2), it is assumed that $\left(\frac{1}{A_z} - \frac{1}{2} \right)$ is positive real, which implies

$$S_{k+1} := \mathbb{E} \left(2 \sum_{i=1}^t \gamma_k \tilde{u}_i \tilde{y}_i \right) \geq 0, \text{ a.s.} \quad (3.34)$$

Adding S_{k+1} to both sides of (3.32) yields

$$\mathbb{E}(V_{k+1} + S_{k+1} | \mathcal{F}_k) \leq V_k + S_k + 2r_v^{-1} \mathbb{E}(\gamma_k \varphi_k^T P_{k+1} \varphi_k). \quad (3.35)$$

Define a new sequence:

$$W_k := \frac{V_k + S_k}{(\ln \mathbb{E}[P_k^{-1}])^c}, \quad c > 1. \quad (3.36)$$

Since $\ln |P_k^{-1}|$ is nondecreasing, it follows from (3.35) and (3.36) that

$$\begin{aligned} \mathbb{E}(W_{k+1}|\mathcal{F}_k) &\leq \frac{V_k + S_k}{(\ln \mathbb{E}|P_{k+1}^{-1}|)^c} + \frac{2r_v^{-1}\mathbb{E}(\gamma_k \varphi_k^T P_{k+1} \varphi_k)}{(\ln \mathbb{E}|P_{k+1}^{-1}|)^c} \\ &\leq W_k + \frac{2r_v^{-1}\mathbb{E}(\gamma_k \varphi_k^T P_{k+1} \varphi_k)}{(\ln \mathbb{E}|P_{k+1}^{-1}|)^c}. \end{aligned} \quad (3.37)$$

According to (3.25) in Lemma 3.4, the martingale convergence theorem (Theorem 3.1) is applicable to (3.37). Therefore, W_k will approach to a finite random variable W_0 almost surely (a.s.), that is, $W_k \rightarrow W_0 < \infty$, a.s., or equivalently,

$$V_k = O\left[(\ln \mathbb{E}|P_k^{-1}|)^c\right], \text{ a.s.}, \quad (3.38)$$

$$S_k = O\left[(\ln \mathbb{E}|P_k^{-1}|)^c\right], \text{ a.s.} \quad (3.39)$$

Applying Lemma 3.1, we have

$$\|\tilde{\theta}_k\|^2 \leq \frac{\tilde{\theta}_k^T P_k^{-1} \tilde{\theta}_k}{\lambda_{\min}(P_k^{-1})} = \frac{V_k}{\lambda_{\min}(P_k^{-1})}. \quad (3.40)$$

It is indicated by (3.23), (3.38) and (3.40) that

$$\|\tilde{\theta}_k\|^2 = O\left[\frac{(\ln \mathbb{E}|P_k^{-1}|)^c}{\lambda_{\min}(P_k^{-1})}\right] = O\left[\frac{(\ln \mathbb{E} r_k)^c}{\lambda_{\min}(P_k^{-1})}\right]. \quad (3.41)$$

This proves (P1.1). If (3.26) instead of (3.25) in Lemma 3.4 is applied in the analysis of the martingale sequences, (P1.2) can be proved in analogy to the above procedure, and thus is omitted here.

Part 2:

We need to show that $\mathbb{E} r_k = O(r_k^o)$ and $\lambda_{\min}((P_k^o)^{-1}) = O[\lambda_{\min}(P_k^{-1})]$ in order to prove (P2.1) and (P2.2).

From (3.33) and (3.39), we have

$$\sum_{i=1}^t \gamma_i \|y_i^o - \varphi_i^T \hat{\theta}_{i+1} - v_i\|^2 = \sum_{i=1}^t \gamma_i \|x_i^o - x_i\|^2 = O[(\ln \mathbb{E} r_k)^c]. \quad (3.42)$$

Define the residual regression vector as $\tilde{\varphi}_k = \varphi_k - \varphi_k^o$. We have

$$\tilde{\varphi}_k = [-x_{k-1} + x_{k-1}^o \quad \cdots \quad -x_{k-n_a} + x_{k-n_a}^o \quad \underbrace{0 \cdots 0}_{n_b+1}]^T.$$

Thus

$$\sum_{i=1}^t \|\tilde{\varphi}_i\|^2 = \sum_{i=1}^t \sum_{j=1}^{n_a} (x_{i-j} - x_{i-j}^o)^2 = O[(\ln \mathbb{E} r_k)^c]. \quad (3.43)$$

Referring to the definition of r_k and r_k^o , we get

$$\begin{aligned}
Er_k &= n_0 p_0^{-1} + r_v^{-1} \sum_{i=1}^{t-1} \mathbb{E} \|\gamma_i \varphi_i\|^2 \\
&\leq n_0 p_0^{-1} + r_v^{-1} \sum_{i=1}^{t-1} \mathbb{E} \|\varphi_i\|^2 \\
&= r_k^o + r_v^{-1} \sum_{i=1}^{t-1} \mathbb{E} \|\tilde{\varphi}_i\|^2 + 2r_v^{-1} \sum_{i=1}^{t-1} \mathbb{E} (\tilde{\varphi}_i^T \varphi_i^o) \\
&\leq 2r_k^o + 2r_v^{-1} \sum_{i=1}^t \mathbb{E} \|\tilde{\varphi}_i\|^2 \\
&= 2r_k^o + O[(\ln Er_k)^c].
\end{aligned}$$

Therefore, Er_k is comparable to r_k^o :

$$Er_k = O(r_k^o). \quad (3.44)$$

From (A3), the boundedness of r_k^o and $\lambda_{\min}((P_k^o)^{-1})$ for $t \geq t_0$ can be determined:

$$n_0 p_0^{-1} + r_v^{-1} \alpha_0 t \leq r_k^o \leq n_0 p_0^{-1} + n_0 r_v^{-1} \beta_0 t^{c_0+1}, \quad (3.45)$$

$$p_0^{-1} + r_v^{-1} \alpha_0 t \leq \lambda_{\min}(P_k^o)^{-1} \leq p_0^{-1} + r_v^{-1} \beta_0 t^{c_0+1}. \quad (3.46)$$

Hence we have

$$\frac{(\ln r_k^o)^c}{\lambda_{\min}((P_k^o)^{-1})} \leq \frac{[\ln(n_0 p_0^{-1} + n_0 r_v^{-1} \beta_0 t^{c_0+1})]^c}{p_0^{-1} + r_v^{-1} \alpha_0 t} \rightarrow 0, \quad (3.47)$$

where $c > 1$. This implies that

$$\ln(r_k^o)^c = o[\lambda_{\min}((P_k^o)^{-1})]. \quad (3.48)$$

Consider an $(n_a + n_b + 1) \times 1$ vector w that has unit norm, i.e., $\|w\| = 1$. Then we have

$$\begin{aligned}
\sum_{i=1}^t (w^T \varphi_i^o)^2 &= \sum_{i=1}^t [w^T (\varphi_i - \tilde{\varphi}_i)]^2 \\
&\leq \sum_{i=1}^t (w^T \varphi_i)^2 + \sum_{i=1}^t (w^T \tilde{\varphi}_i)^2 \\
&= \sum_{i=1}^t (w^T \varphi_i)^2 + O[(\ln r_k^o)^c].
\end{aligned}$$

It can be readily obtained that

$$\begin{aligned}
\lambda_{\min}((P_k^o)^{-1}) &\leq \lambda_{\min}(P_k^{-1}) + O[(\ln r_k^o)^c] \\
&= \lambda_{\min}(P_k^{-1}) + o[\lambda_{\min}((P_k^o)^{-1})].
\end{aligned}$$

Thus

$$\lambda_{\min}((P_k^o)^{-1}) = O[\lambda_{\min}(P_k^{-1})]. \quad (3.49)$$

Now the convergence property of $\hat{\theta}_k$ can be written in the following form by summarizing (3.41), (3.44) and (3.49):

$$\|\hat{\theta}_k - \theta\|^2 = O\left[\frac{(\ln r_k^o)^c}{\lambda_{\min}((P_k^o)^{-1})}\right]. \quad (3.50)$$

The conclusion (P2.1) is proven. Similarly, (P2.2) can be proved in light of (P1.2).

Part 3:

Now from (3.47) and (3.50), we have

$$\begin{aligned} \|\hat{\theta}_k - \theta\|^2 &= O\left[\frac{[\ln(n_0 p_0^{-1} + n_0 r_v^{-1} \beta_0 t^{c_0+1})]^c}{p_0^{-1} + r_v^{-1} \alpha_0 t}\right] \\ &= O\left[\frac{(\ln t)^c}{t}\right] \rightarrow 0, \quad \text{a.s., } c > 1. \end{aligned} \quad (3.51)$$

This proves (C1). The conclusion (C2) can be derived similarly from (P2.2). This completes the proof. \square

Remark 3.2. Assumptions (A1) and (A2) show that v_k is an independent noise sequence with zero mean and bounded time-varying variance.

Remark 3.3. The assumption (A3) refers to the persistent excitation (PE) condition that is a standard assumption; its practical meaning is to have rich enough excitation signals to drive and further identify the system.

Remark 3.4. For systems with complete input/output measurements, under the stochastic framework, the convergence analysis of least squares identification algorithms and gradient based algorithms have been discussed extensively [17; 18; 20; 32; 8]. In Theorem 4.1, following the similar line, the results have been extended to the convergence properties of the modified Kalman filter based parameter estimation algorithm for systems with randomly missing measurements in a TCP-like network environment. It is worthwhile noting that γ_k is involved in the developed algorithm to characterize the random missing measurements, which also poses challenges on the proof of Theorem 4.1; in this sense, the proof procedure is different from existing results in terms of incorporating the randomly data missing into account.

Remark 3.5. Theorem 3.2 reveals that the parameter estimation error of the algorithm (3.13)-(3.17), even in the presence of output missing, will converge to zero at the speed of $O[(\ln k)^c/k]$.

3.4.3 Convergence of Output Estimation

It is also important to analyze the convergence properties of the output estimation. To establish this, we give the next theorem.

Theorem 3.3. Assume that the assumptions (A1)-(A4) hold, and

$$(A5) \text{ The input is bounded, i.e., } u_k^2 < \infty \text{ for any } k.$$

Then there exists a positive integer t_0 such that for any $k \geq k_0$ the output estimation error $z_k - y_k$ satisfies

$$(C3) \quad \sum_{i=k_0}^k (z_i - y_i^o)^2 = O[(\ln k)^{c+1}], \quad \text{a.s., } c > 1,$$

$$(C4) \quad \frac{1}{k} \sum_{i=t_0}^k (z_i - y_i^o)^2 = O\left[\frac{(\ln k)^{c+1}}{k}\right] \rightarrow 0, \quad \text{a.s., } c > 1.$$

Proof. First, we have

$$\begin{aligned} z_k - y_k^o &= (1 - \gamma_k)(\hat{y}_k - y_k^o) \\ &= (1 - \gamma_k)(\varphi_k^T \hat{\theta}_k - y_k^o) \\ &= (1 - \gamma_k)(\varphi_k^T \hat{\theta}_k - x_k - v_k). \end{aligned}$$

Then it follows that

$$\begin{aligned} E(z_k - y_k^o) &= E\left[(1 - \gamma_k)(\varphi_k^T \hat{\theta}_k - x_k^o - v_k)\right] \\ &= E\left[(1 - \gamma_k)(\varphi_k^T \hat{\theta}_k - x_k^o)\right] + \gamma_k r_v \\ &= E\left[(1 - \gamma_k)(x_k - r_v^{-1} P_{k+1} \varphi_k e_k - x_k^o)\right] + \gamma_k r_v \\ &= E\left[(1 - \gamma_k)(x_k - x_k^o)\right] + \gamma_k r_v. \end{aligned}$$

According to Theorem 3.2, $\|\tilde{\theta}_k\|^2$ converges to 0 as k increases. With the assumption of input being bounded, there must be a positive integer k_0 such that for any $k \geq k_0$ the regression vector is bounded if the system is stable (as assumed in (A4)). That is, $\exists k_0 \in N^+$ and $\varepsilon > 0$,

$$\|\varphi_k\|^2 \leq \varepsilon < \infty \quad \text{for } k \geq k_0.$$

Then

$$\begin{aligned}
\sum_{i=k_0}^k (z_i - y_i^o)^2 &= \sum_{i=k_0}^k \mathbb{E} [(1 - \gamma_k)(x_i - x_i^o)] + \gamma r_v \\
&= O \left(\sum_{i=k_0}^k \|\varphi_i^T \tilde{\theta}_i\|^2 \right) \\
&= O \left(\sum_{i=k_0}^k \varepsilon \|\tilde{\theta}_i\|^2 \right) \\
&= O \left[\sum_{i=k_0}^k \varepsilon \frac{(\ln i)^c}{i} \right] \\
&= O [(\ln k)^{c+1}].
\end{aligned}$$

This proves (C3). The conclusion (C4) can be achieved by dividing k on both sides of the above equation. This completes the proof. \square

Remark 3.6. The proposed output estimator possesses a simple structure, yet it is effective: The estimation error is proven to converge to zero in average sense at the speed of $O [(\ln k)^{c+1}/k]$.

3.5 Numerical Examples

In this section, the proposed algorithm is examined through simulation studies.

The proposed algorithm (3.13)-(3.17) is applied to the input-output data collected from a second-order single-input-single-output (SISO) plant placed in a network environment:

$$\begin{aligned}
y_k^o &= \frac{b_0 + b_1 z^{-1}}{1 + a_1 z^{-1} + a_2 z^{-2}} u_k^o + v_k, \\
u_k^o &= \lambda_k u_k, \\
y_k &= \gamma_k y_k^o.
\end{aligned}$$

The desired input $\{u_k\}$ is a uniformly-distributed sequence with zero mean and unit variance. As aforementioned, due to the network-induced randomly data missing, the actual input to the plant is $\{u_k^o\}$, an intermittent version of $\{u_k\}$. In a similar way, $\{y_k^o\}$ is the output response of the plant, yet it is $\{y_k\}$ that is finally received and used for identification. $\{v_k\}$ is a white noise sequence with zero mean and variance r_v . The parameter vector $\theta = [a_1 \ a_2 \ b_0 \ b_1]^T$ is to be estimated. Here, θ is supposed to be

$$\theta = [0.523 \ 0.349 \ 0.440 \ 0.762]^T.$$

In the following simulation studies, we carry out experiments for three different cases regarding the data completeness and the data missing pattern.

Example 1: $\lambda = 0.8$ and $\gamma = 0.7$. In this case, about 20% of the input data and 30% of the output data are missing.

The estimated parameters and corresponding estimation errors of four parameter unknowns are shown in Fig. 3.2 and Table 1, respectively. It is observed that all the parameter estimates gradually converge to their true values as k increases.

To further quantify the estimation accuracy, define the relative parameter estimation error as

$$\delta_{par}\% = \frac{\|\hat{\theta}_k - \theta\|}{\|\theta\|} \times 100\%.$$

It is shown in Fig. 3.3 that δ_{par} has a clear tendency to approach zero. To examine the output estimation performance, a comparison between the estimated outputs and true outputs during the time range $501 \leq k \leq 550$ is illustrated in Fig. 3.4: The dashed lines illustrate time instants when data missing occurs, and corresponding small asterisks represent the estimated outputs at these time instants. In addition, define the average output estimation error

$$\delta_{out} = \frac{1}{k} \sum_{i=1}^k (z_i - y_i^o)^2.$$

The curve of the average output estimation error is shown in Fig. 3.5. From both Fig. 3.4 and Fig. 3.5, we can observe that the output estimation also exhibits good performance.

Table 3.1: Example 1: Intermediate parameter estimates and estimation errors ($\lambda = 0.8$ and $\gamma = 0.7$).

k	\hat{a}_1	\hat{a}_2	\hat{b}_0	\hat{b}_1	$\delta_{par}(\%)$
10	0.567627	1.6701	3.20898	1.68664	296.32
63	0.42823	0.174088	0.49568	0.631873	22.5757
125	0.46955	0.22639	0.474952	0.726835	13.1901
250	0.506762	0.278517	0.417336	0.737531	7.36484
500	0.540054	0.308427	0.407041	0.747389	5.26075
1000	0.53704	0.335451	0.428435	0.749043	2.41542
True values	0.523	0.349	0.440	0.762	

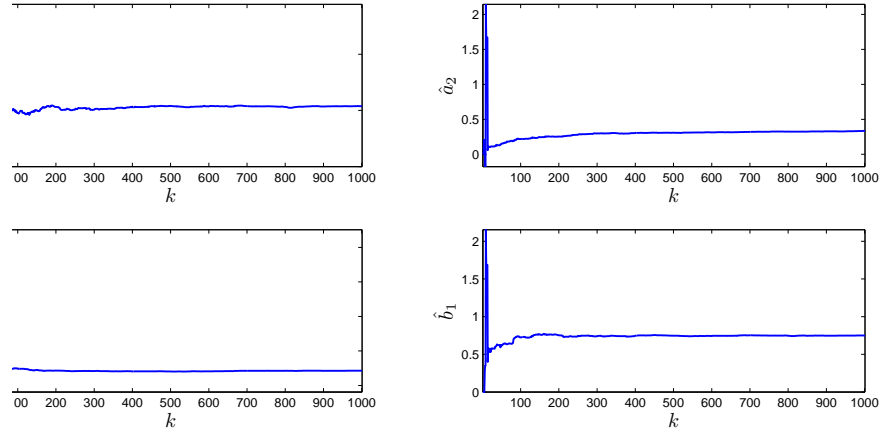


Figure 3.2: Example 1: Estimates of the parameter unknowns a_1 , a_2 , b_0 and b_1 ($\lambda = 0.8$ and $\gamma = 0.7$).

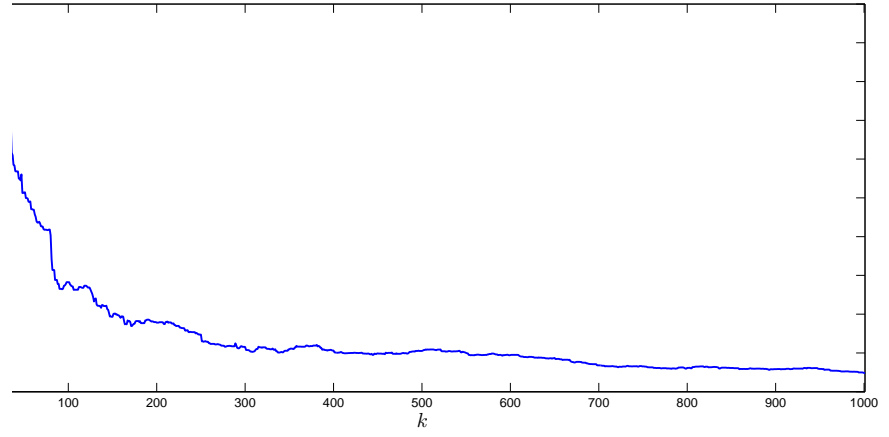


Figure 3.3: Example 1: Relative parameter estimation error versus time ($\lambda = 0.8$ and $\gamma = 0.7$).

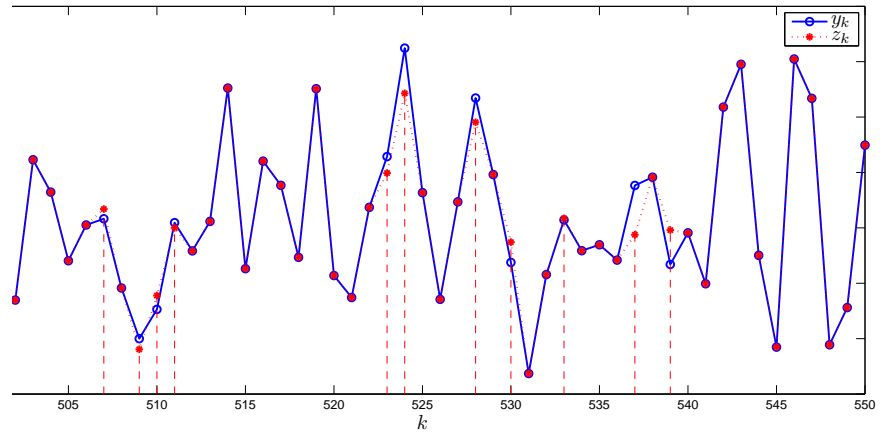


Figure 3.4: Example 1: Comparison between estimated and true outputs ($\lambda = 0.8$ and $\gamma = 0.7$).

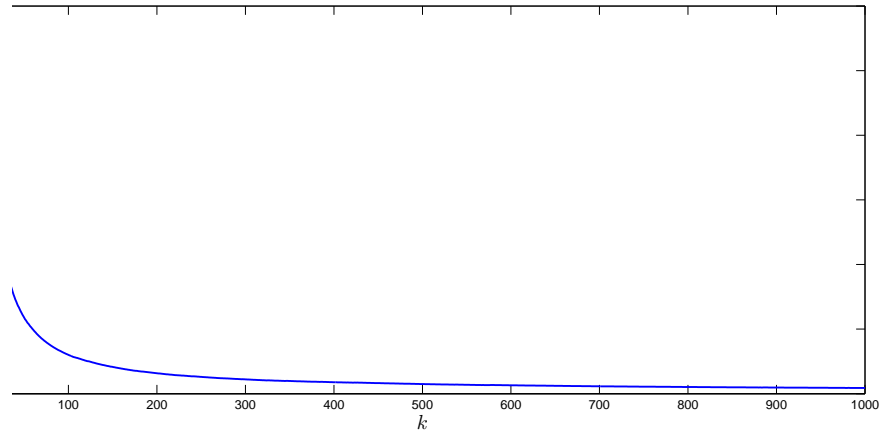


Figure 3.5: Example 1: Average output estimation error versus time ($\lambda = 0.8$ and $\gamma = 0.7$).

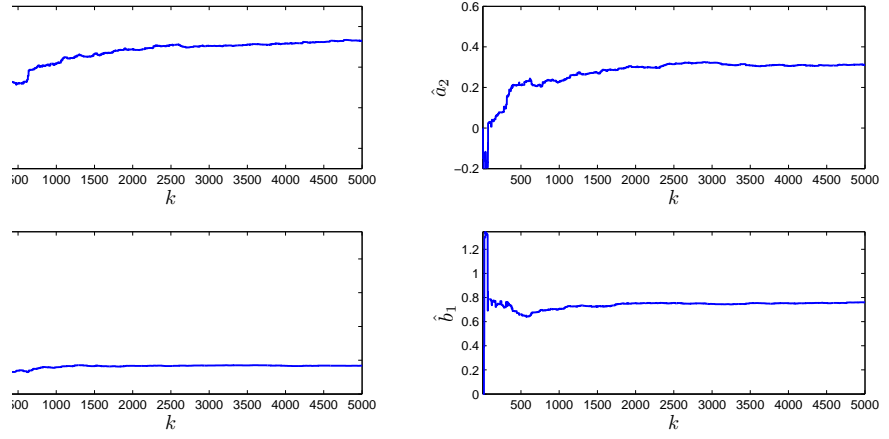


Figure 3.6: Example 2: Estimates of the parameter unknowns a_1 , a_2 , b_0 and b_1 ($\lambda = 0.4$ and $\gamma = 0.2$).

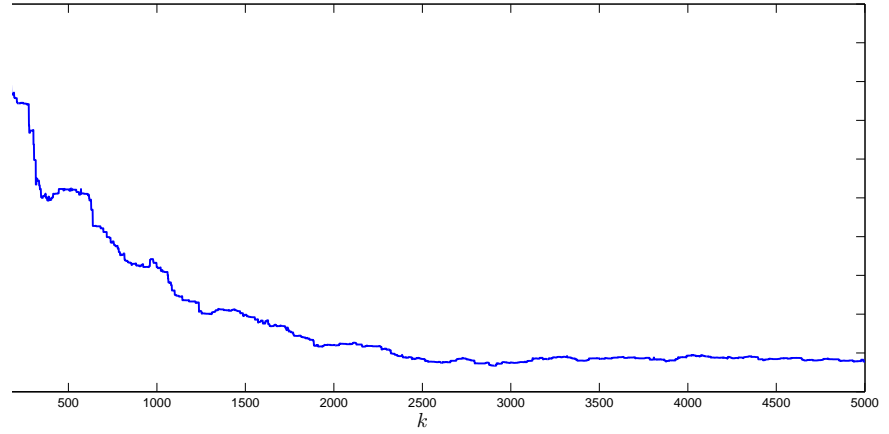


Figure 3.7: Example 2: Relative parameter estimation error versus time ($\lambda = 0.4$ and $\gamma = 0.2$).

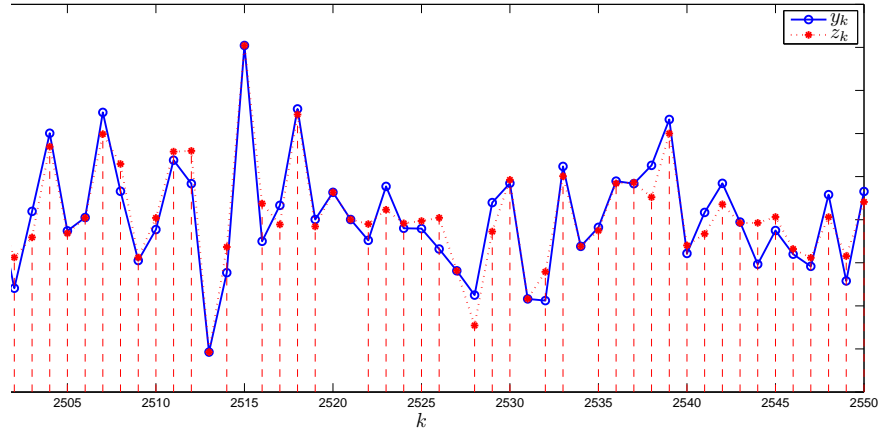


Figure 3.8: Example 2: Comparison between estimated and true outputs ($\lambda = 0.4$ and $\gamma = 0.2$).

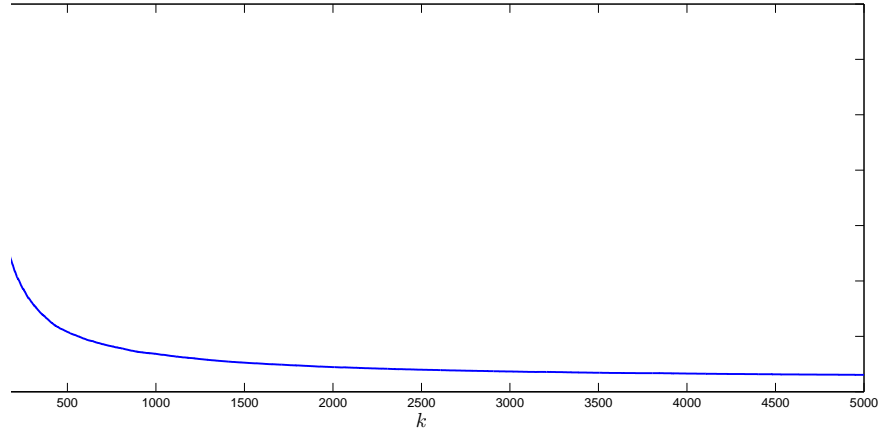


Figure 3.9: Example 2: Average output estimation error versus time ($\lambda = 0.4$ and $\gamma = 0.2$).

Table 3.2: Example 2: Intermediate parameter estimates and estimation errors ($\lambda = 0.4$ and $\gamma = 0.2$).

k	\hat{a}_1	\hat{a}_2	\hat{b}_0	\hat{b}_1	$\delta_{par}(\%)$
10	-0.182577	-0.148257	0	0	113.976
313	0.299781	0.115871	0.42551	0.762816	29.875
625	0.338345	0.239968	0.325255	0.65233	24.6696
1250	0.452098	0.269698	0.424789	0.72987	10.371
2500	0.509494	0.314595	0.414112	0.755212	4.21955
5000	0.530625	0.312564	0.420325	0.760732	3.8951
True values	0.523	0.349	0.440	0.762	

Example 2: $\lambda = 0.4$ and $\gamma = 0.2$. In the second case, about 60% of the input data and 80% of the output data are missing, respectively. It is clear that in this example, the missing data scenario is much worse than that in Example 1.

Estimates of four parameters are shown in Fig. 3.6 and Table 2, respectively. Even though the available output measurements are more scarce than those in Example 1, it is still observed that all the parameter estimates gradually converge to their true values as k increases.

The relative estimation error, $\delta_{par}\%$, shown in Fig. 3.7, is still approaching to zero. Performance of the output estimator in terms of both the difference from the true outputs and average output estimation error is illustrated in Fig. 3.8 and Fig. 3.9 (dashed red curve), respectively.

By comparing Fig. 3.3 to Fig. 3.7, and Fig. 3.5 to Fig. 3.9, we note that: (1) The estimation performance in Example 1 is better than that in Example 2, because there were less data missed in Example 1; (2) to achieve the same level of estimation accuracy, more data would be needed in Example 2, when more measurements are missing in this case; (3) the estimation performance depends on the data completeness that can be characterized by both λ and γ .

Example 3: Different data missing patterns. It is also paramount to explore the influence of missing data patterns, e.g., $\{\lambda_k\}$ and $\{\gamma_k\}$, on the estimation performance. In

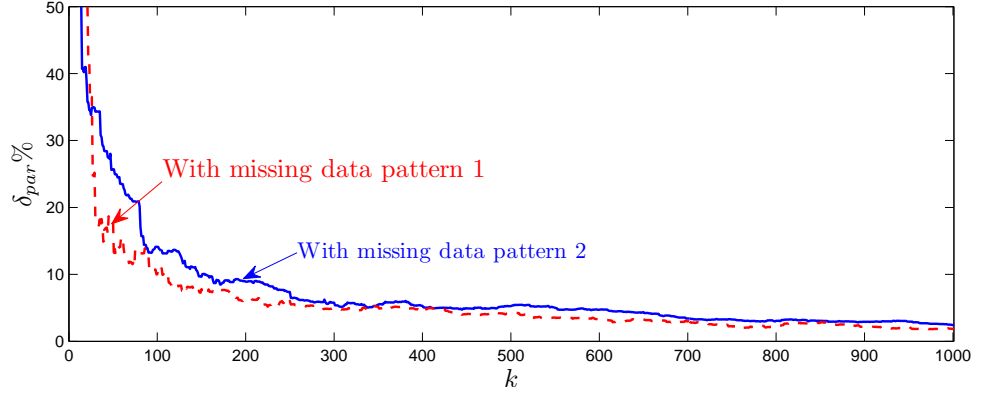


Figure 3.10: Example 3: Relative parameter estimation errors for two cases with different data missing patterns ($\lambda = 0.8$ and $\gamma = 0.7$).

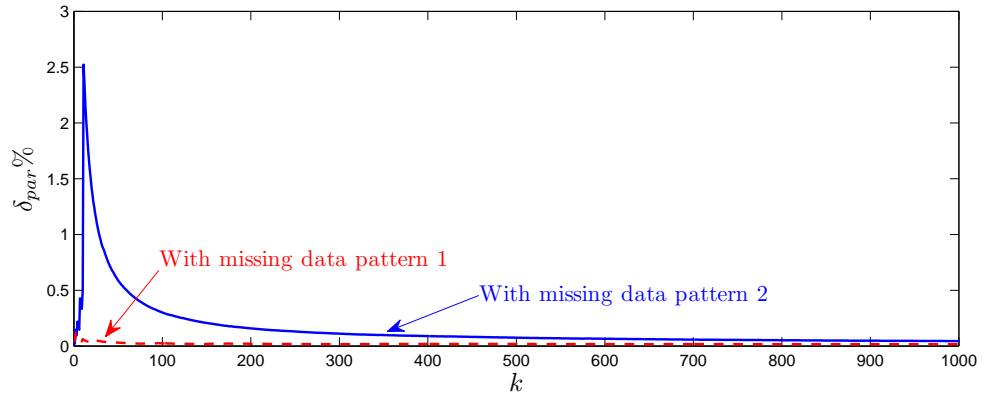


Figure 3.11: Example 3: Average output estimation errors for two cases with different data missing patterns ($\lambda = 0.8$ and $\gamma = 0.7$).

this example, we implement the proposed algorithm twice with same $\lambda = 0.8$ and $\gamma = 0.7$, but the input and output availability sequences λ_k and $\{\gamma_k\}$ s are randomly generated and thus different.

The resulting relative parameter estimation errors and average output estimation errors are visually compared in Fig. 3.10 and Fig. 3.11. Obviously, even with identical λ and γ , estimation performance may still vary. In fact, not only the data completeness, but also the missing data patterns are significant factors for identification of systems with randomly missing measurements in a network environment.

3.6 Summary

In this chapter, we have studied the problem of parameter estimation of systems placed in a TCP-based network environment. For such a problem, missing input and output data is the primary concern as data transmitted over a network may encounter time delays or even packet loss. Random input and output missing are modeled as two Bernoulli processes. A missing output estimator is designed, and further a modified Kalman filter based recursive estimation algorithm is developed. Convergence properties for both parameter estimation and output estimation are established. Simulation examples verify the effectiveness of the proposed algorithm and also illustrate that the data completeness and the data missing pattern would affect the estimation performance. It is worthwhile noting that the proposed algorithm can handle two practical cases: (1) The input and output have different probabilities of missing data; (2) the probabilities of missing data are larger than $1/2$. Thus, the design in this work makes an important step forward in addressing practical issues of network-induced randomly missing data for identification of systems over lossy networks.

It is worth noting that the idea could be extended to other widely applied communication protocols, such as UDP, Profibus, factory instrumentation protocol (FIP) [39] and so on. In addition, although the orders of the system model are assumed to be fixed and known here, the framework could be extended to the case where model orders are also needed to be identified. Moreover, it is desirable to further develop adaptive control schemes for NCSs based on the proposed parameter estimation algorithms. These topics are worth further studying.

CHAPTER 4

ADAPTIVE CONTROL FOR NETWORKED SYSTEMS

4.1 Introduction and Literature Review

Control design has long been a primary concern in the field of NCSs. As aforementioned, the introduction of networks also presents some challenges such as the *limited feedback information* caused by packet transmission delays and packet loss; both of them are due to the sharing and competition of the transmission medium. The information transmission delay arises from by the limited capacity of the communication network used in a control system, whereas the packet loss is caused by the unavoidable data losses or transmission errors. Both the information transmission delay and packet loss may result in randomly missing output measurements at the controller node, as shown in Fig. 4.1. Obviously, randomness of available output measurements brings difficulties for control analysis and design for NCSs. Yet consequently, we note that adaptive control schemes, which are continuously adapting the control signals to the environment, have great potentials in NCSs. In this chapter, we would pursue this topic to develop adaptive controllers for NCSs. Before proceeding further, we shall review related literature exhaustively to gain a panoramic view.

Limited feedback information (information transmission delays and packet losses) can degrade the performance of systems or even cause instability. Various methodologies have been proposed for modeling, stability analysis, and controller design for NCSs in the presence of limited feedback information. A novel feedback stabilization solution of multiple coupled control systems with limited communication is proposed by bringing together communication and control theoretical issues in [40]. Further the control and communication co-design methodology is applied in [88; 37] – a method of stabilizing linear NCSs with medium access constraints and transmission delays by designing a delay-compensated feedback controller and an accompanying medium access policy is presented. In [90], the

relationship of sampling time and maximum allowable transfer interval to keep the systems stable is analyzed by using a stability region plot; the stability analysis of NCSs is addressed by using a hybrid system stability analysis technique. In [77], a new NCS protocol, try-once-discard (TOD), which employs dynamic scheduling method, is proposed and the analytic proof of global exponential stability is provided based on Lyapunov's second method. In [3], the conditions under which NCSs subject to dropped packets are mean square stable are provided. Output feedback controller that can stabilize the plant in the presence of delay, sampling, and dropout effects in the measurement and actuation channels is developed in [57]. In [85], the authors model the NCSs with packet dropout and delays as ordinary linear systems with input delays and further design state feedback controllers using Lyapunov-Razumikhin function method for the continuous-time case, and Lyapunov-Krasovskii based method for the discrete-time case, respectively. In [86], the time delays and packet dropout are simultaneously considered for state feedback controller design based on a delay-dependent approach; the maximum allowable value of the network-induced delays can be determined by solving a set of linear matrix inequalities (LMIs). Most recently, Gao, *et. al.*, for the first time, incorporate simultaneously three types of communication limitation, e.g., measurement quantization, signal transmission delay, and data packet dropout into the NCS design for robust \mathcal{H}_∞ state estimation [25], and passivity based controller design [27], respectively. Further, a new delay system approach that consists of multiple successive delay components in the state, is proposed and applied to network-based control in [28].

However, the results obtained for NCSs are still limited: Most of the aforementioned results assume that the plant is given and model parameters are available, while few papers address the analysis and synthesis problems for NCSs whose plant parameters are *unknown*. In fact, while controlling a real plant, the designer rarely knows its parameters accurately [58]. To the best of the our knowledge, adaptive control for systems with unknown parameters and randomly missing outputs in a network environment has not been fully investigated, which is the focus of this paper.

It is worth noting that systems with *regular* missing outputs – a special case of those with randomly missing outputs – can also be viewed as multirate systems which have uniform but various input/output sampling rates [10]. Such systems may have regular-output-missing feature. In [17], Ding, *et. al.* use an auxiliary model and a modified

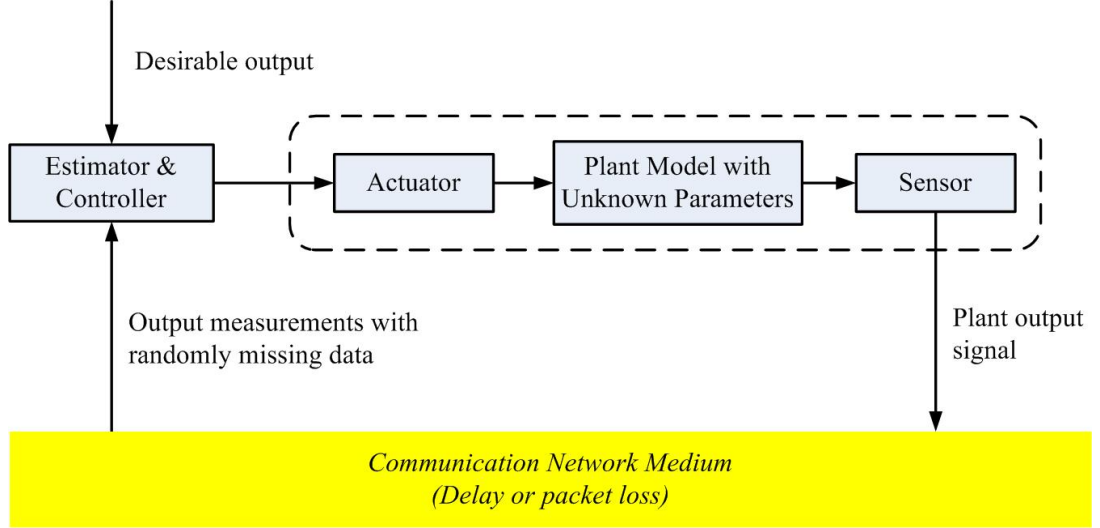


Figure 4.1: An NCS with randomly missing outputs.

recursive least squares (RLS) algorithm to realize simultaneous parameter and output estimation of dual-rate systems. Further, a least squares based self-tuning control scheme is studied for dual-rate linear systems [18] and nonlinear systems [19], respectively. However, network-induced limited feedback information unavoidably results in *randomly* missing output measurements. To generalize and extend the adaptive control approach for multirate systems [18; 19] to NCSs with randomly missing output measurements and unknown model parameters is another motivation of this work.

In this chapter, we first model the availability of output as a Bernoulli process. Then we design an output estimator to online estimate the missing output measurements, and further propose a novel Kalman filter based method for parameter estimation with randomly output missing (Please note that the method is from Chapter 3 but slightly different). Based on the estimated output or the available output, and the estimated model parameters, an adaptive control is proposed to make the output track the desired signal. Convergence of the proposed output estimation and adaptive control algorithms is analyzed.

4.2 Problem Formulation

The problem of interest in this work is to design an adaptive control scheme for networked systems with unknown model parameters and randomly missing outputs. In Fig. 4.1,

the output measurements y_k could be unavailable at the controller node at some time instants because of the network-induced limited feedback information, e.g., transmission delay and/or packet loss. The data transmission protocols like TCP guarantee the delivery of data packets in this way: When one or more packets are lost the transmitter retransmits the lost packets. However, since a retransmitted packet usually has a long delay that is not desirable for control systems, the retransmitted packets are outdated by the time they arrive at the controller [3; 38]. Therefore, in this paper, it is assumed that the output measurements that are delayed in transmission are regarded as missed ones.

The availability of y_k can be viewed as a random variable γ_k . γ_k is assumed to have Bernoulli distribution:

$$\mathbb{E}(\gamma_k \gamma_s) = \mathbb{E}\gamma_k \mathbb{E}\gamma_s \text{ for } k \neq s, \quad \mathbb{P}(\gamma_k) = \begin{cases} \gamma, & \text{if } \gamma_k = 1, \\ 1 - \gamma, & \text{else if } \gamma_k = 0, \end{cases} \quad (4.1)$$

where $0 < \gamma \leq 1$.

Consider a SISO process described by an output-error (OE) model:

$$A_z x_k = B_z u_k, \quad y_k = x_k + v_k, \quad (4.2)$$

where u_k is the system input, y_k the output and v_k the disturbing white noise with variance r_v . A_z and B_z are two backshift polynomials defined as

$$\begin{aligned} A_z &= 1 + a_1 z^{-1} + a_2 z^{-2} + \cdots + a_{n_a} z^{-n_a}, \\ B_z &= b_0 + b_1 z^{-1} + b_2 z^{-2} + \cdots + b_{n_b} z^{-n_b}. \end{aligned}$$

The polynomial orders n_a and n_b are assumed to be given. Eqn. (4.2) can be written equivalently as the following linear regression model:

$$y_k = \varphi_k^{o\top} \theta + v_k, \quad (4.3)$$

where

$$\begin{aligned} \varphi_k^o &= [-x_{k-1} \ -x_{k-2} \ \cdots \ -x_{k-n_a} \ u_k \ u_{k-1} \ \cdots \ u_{k-n_b}]^\top, \\ \theta &= [a_1 \ a_2 \ \cdots \ a_{n_a} \ b_0 \ b_1 \ \cdots \ b_{n_b}]^\top. \end{aligned}$$

Vector φ_k^o represents system's excitation and response information necessary for parameter estimation, while vector θ contains model parameters to be estimated.

For a system with the OE model placed in a networked environment subject to randomly missing outputs, our objectives are:

- P1. Design an output estimator to online estimate the missing output measurements.
- P2. Develop a recursive Kalman filter based identification algorithm to estimate unknown model parameters.
- P3. Propose an adaptive tracking controller to make the system output track a given desired signal.
- P4. Analyze the convergence properties of the proposed algorithms.

4.3 Derivation of the Algorithm

There are two main challenges of the adaptive control design for a networked system as depicted in Fig. 4.1: (1) randomly missing output measurements; (2) unknown system model parameters. Therefore, in this section, we first propose algorithms for missing output estimation and unknown model parameter estimation, and then design the adaptive control scheme.

4.3.1 Parameter Estimation and Missing Output Estimation

Consider the model in (4.3). It is shown by [32] and [4] that the corresponding Kalman filter can be conveniently used for parameter estimation. In combination with an auxiliary model, the Kalman filter based parameter estimation algorithm for an OE model is given by

$$\hat{\theta}_k = \hat{\theta}_{k-1} + K_a^k (y_k - \varphi_k^{aT} \hat{\theta}_{k-1}), \quad (4.4)$$

$$K_a^k = \frac{P_{k-1}^a \varphi_k^a}{r_v + \varphi_k^{aT} P_{k-1}^a \varphi_k^a}, \quad (4.5)$$

$$P_k^a = P_{k-1}^a - \frac{P_{k-1}^a \varphi_k^a \varphi_k^{aT} P_{k-1}^a}{r_v + \varphi_k^{aT} P_{k-1}^a \varphi_k^a}, \quad (4.6)$$

$$x_k^a = \varphi_k^{aT} \hat{\theta}_k, \quad (4.7)$$

$$\varphi_k^a = [-x_{k-1}^a \quad -x_{k-2}^a \quad \cdots \quad -x_{k-n_a}^a \quad u_k \quad u_{k-1} \quad \cdots \quad u_{k-n_b}]^T, \quad (4.8)$$

where $\hat{\theta}_k$ represents the estimated parameter vector at instant k .

It is worth to note that the above algorithm as shown in (4.4)-(4.8) cannot be *directly* applied to the parameter estimation of systems with randomly missing outputs in a network environment, as y_k in (4.4) may not be available. This motivates us to develop a new

algorithm that can simultaneously online estimate the unavailable missing output and estimate system parameters under the network environment. The proposed algorithm consists of two steps.

Step 1: Output estimation

Albertos, *et. al.* propose a simple algorithm that uses the input-output model, replacing the unknown past values by estimates when necessary [1]. Inspired by this work, we design the following output estimator:

$$z_k = \gamma_k y_k + (1 - \gamma_k) \hat{y}_k, \quad (4.9)$$

with

$$\hat{y}_k = \varphi_k^T \hat{\theta}_{k-1}.$$

In (4.9), γ_k is a Bernoulli random variable used to characterize the availability of y_k at time instant k at the controller node, as defined in (4.1). With the time-stamp technique, the controller node can detect the availability of the output measurements, and thus, the values of γ_k s (either 1 or 0) are known. The knowledge of their corresponding probability γ s is not used in the designed estimator. The structure of the designed output estimator is intuitive and simple yet very effective, which will be seen soon from the simulation examples.

Step 2: Model parameter estimation

Replacing y_k in the algorithm (4.4)-(4.8) by z_k , defining a new φ_k , respectively, and considering the random variable γ_k , we readily obtain the following algorithm:

$$\hat{\theta}_k = \hat{\theta}_{k-1} + K_k(z_k - \varphi_k^T \hat{\theta}_{k-1}), \quad (4.10)$$

$$K_k = \frac{P_{k-1} \varphi_k}{r_v + \varphi_k^T P_{k-1} \varphi_k}, \quad (4.11)$$

$$P_k = P_{k-1} - \gamma_k \frac{P_{k-1} \varphi_k \varphi_k^T P_{k-1}}{r_v + \varphi_k^T P_{k-1} \varphi_k}, \quad (4.12)$$

$$x_{b,k} = \varphi_k^T \hat{\theta}_k, \quad (4.13)$$

$$\varphi_k = [-x_{b,k-1} \ -x_{b,k-2} \ \cdots \ -x_{b,k-n_a} \ u_k \ u_{k-1} \ \cdots \ u_{k-n_b}]^T. \quad (4.14)$$

Remark 4.1. Consider two extreme cases. If the availability sequence $\{\gamma_1, \dots, \gamma_k\}$ constantly assumes 1, then no output measurement is lost, and the algorithm above will reduce to the algorithm (4.4)-(4.6). On the other hand, if the availability sequence $\{\gamma_k\}$ constantly takes 0, then all output measurements are lost, and the parameter estimates just preserve the initial values.

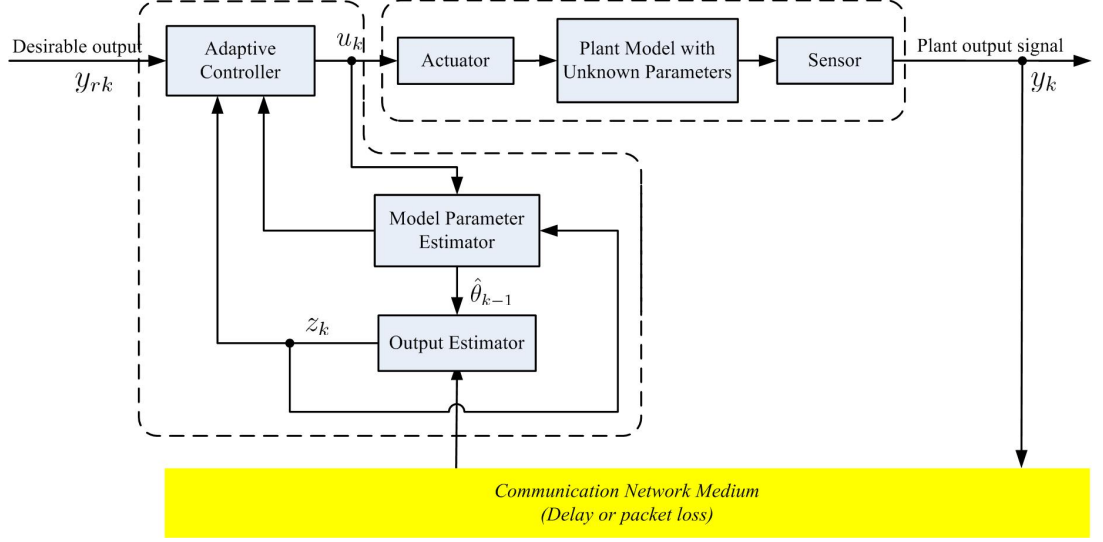


Figure 4.2: Adaptive control diagram.

4.3.2 Adaptive Control Design

Consider the tracking problem. Let $y_{r,k}$ be a desired output signal, and define the output tracking error

$$\zeta_k := y_k - y_{r,k}.$$

If the control law u_k is appropriately designed such that $y_{r,k} = \varphi_k^o \theta$, then the tracking error ζ_k approaches zero finally. Replacing θ by $\hat{\theta}_{k-1}$ and φ_k^o by φ_k yields

$$\begin{aligned} y_{r,k} &= \varphi_k^T \hat{\theta}_{k-1} = - \sum_{i=1}^{n_a} \hat{\theta}_{i,k-1} z_{k-i} + \sum_{i=0}^{n_b} \hat{\theta}_{n_a+i+1,k-1} u_{k-i} \\ &= -\hat{a}_{1,k-1} z_{k-1} - \cdots - \hat{a}_{n_a,k-1} z_{k-n_a} + \hat{b}_{0,k-1} u_k + \cdots + \hat{b}_{n_b,k-1} u_{k-n_b}. \end{aligned}$$

Therefore, the control law can be designed as

$$u_k = \frac{1}{\hat{b}_{0,k-1}} \left[y_{r,k} + \sum_{i=1}^{n_a} \hat{a}_{i,k-1} z_{k-i} - \sum_{i=1}^{n_b} \hat{b}_{i,k-1} u_{k-i} \right]. \quad (4.15)$$

The proposed adaptive control scheme consists of the missing output estimator [Equation (4.9)], model parameter estimator [Equations (4.10-4.14)], and the adaptive control law [Equation (4.15)]. The overall control diagram is shown in Fig. 4.2.

4.4 Convergence Analysis

This section focuses on the analysis of some convergence properties. Some preliminaries are first summarized to facilitate the following convergence analysis of parameter estimation

in (4.10)-(4.12) and of output estimation in (4.9). Inspired by [8], [19] and [17], the convergence analysis is carried out under the stochastic framework.

4.4.1 Preliminaries

To facilitate the convergence analysis, directly applying the matrix inversion formula [35]

$$(A + BCD)^{-1} = A^{-1} - A^{-1}B(C^{-1} + DA^{-1}B)^{-1}DA^{-1},$$

the proposed parameter estimation algorithm in Section 4.3.1 [(4.10)-(4.12)] can be equivalently rewritten as:

$$\hat{\theta}_k = \hat{\theta}_{k-1} + r_v^{-1} P_k \varphi_k (z_k - \varphi_k^T \hat{\theta}_{k-1}), \quad (4.16)$$

$$P_k^{-1} = P_{k-1}^{-1} + r_v^{-1} \gamma_k \varphi_k \varphi_k^T. \quad (4.17)$$

Suppose that P_k is initialized by $p_0 I$, where p_0 is a positive real value large enough, and define $r_k = \text{tr}(P_k^{-1})$. The relation between r_k and $|P_k^{-1}|$ can be established in the following lemma.

Lemma 4.1. The following relation holds:

$$\ln \mathbb{E}|P_k^{-1}| = O(\ln \mathbb{E} r_k). \quad (4.18)$$

Proof. The proof is the same with that of Lemma 4.1 in Chapter 3. \square

The next lemma shows the convergence of two infinite series that will be useful later.

Lemma 4.2. The following inequalities hold:

$$\sum_{i=1}^t \mu_i r_v^{-1} \mathbb{E}(\varphi_i^T P_i \varphi_i) \leq \ln \mathbb{E}|P_k^{-1}| + n_0 \ln p_0 \text{ a.s.} \quad (4.19)$$

$$\sum_{i=1}^{\infty} \mu_i r_v^{-1} \frac{\mathbb{E}(\varphi_i^T P_i \varphi_i)}{(\ln \mathbb{E}|P_i^{-1}|)^c} < \infty \text{ a.s.}, \quad (4.20)$$

where $c > 1$.

Proof: The proof can be done along the similar way as Lemma 2 in [18] and is omitted here. \square

4.4.2 Convergence Analysis

To carry out the convergence analysis of the proposed algorithms, it is essential to appropriately construct a martingale process satisfying the conditions of Theorem 3.1. Main results on the convergence properties of the proposed algorithm are summarized in the following Theorem.

Theorem 4.1. For the system considered in (4.3), assume that

(A1) $\{v_k, \mathcal{F}_k\}$ is a martingale difference sequence satisfying

$$\mathbb{E}(v_k | \mathcal{F}_{k-1}) = 0, \quad \text{a.s.} \quad (4.21)$$

$$\mathbb{E}(v_k^2 | \mathcal{F}_{k-1}) = r_v < \infty, \quad \text{a.s.} \quad (4.22)$$

(A2) $\frac{1}{A_z} - \frac{1}{2}$ is strictly positive real,

(A3) B_z is stable, i.e., zeros of B_z are inside the closed unit disk.

Suppose the desired output signal is bounded: $|y_{r,k}| < \infty$. Applying the missing output estimator [Equation (4.9)], model parameter estimator [Equations (4.10-4.14)], and the adaptive control law [Equation (4.15)], then the output tracking error has the property of minimum variance, i.e.,

$$\begin{aligned} (1) \quad & \lim_{k \rightarrow \infty} \frac{1}{k} \sum_{i=1}^k (y_{r,i} - y_i + v_i)^2 = 0, \quad \text{a.s.}; \\ (2) \quad & \lim_{k \rightarrow \infty} \frac{1}{k} \sum_{i=1}^k \mu_i \mathbb{E} \{ (z_i - y_{r,i})^2 | \mathcal{F}_{i-1} \} = r_v < \infty, \quad \text{a.s.} \end{aligned}$$

Proof: As pointed out in [8] and [31], from (A2) it follows that

$$\frac{1}{k} \sum_{i=1}^k u_i^2 \leq O(1) + O\left(\frac{c_1}{k} \sum_{i=1}^k y_i^2\right), \quad \text{a.s.} \quad (4.23)$$

Here, c_1 is a positive constant. Define the following vectors:

$$e_k = z_k - \varphi^T \hat{\theta}_{k-1},$$

$$\bar{\eta}_k = y_k - x_{b,k},$$

$$\eta_k = \gamma_k \bar{\eta}_k,$$

$$\bar{\tau}_k = y_{r,k} - y_k + v_k,$$

$$\tau_k = \gamma_k \bar{\tau}_k.$$

From (4.2), (4.3), (4.16) and (4.16), it follows that

$$\eta_k = \gamma_k(x_k - x_{b,k} + v_k), \quad (4.24)$$

$$\eta_k = (1 + r_v^{-1}\varphi_k^T P_{k-1}\varphi_k)^{-1}e_k, \quad (4.25)$$

$$e_k = -\tau_k + \gamma_k v_k. \quad (4.26)$$

Also define the parameter estimation error vector and a Lyapunov-like function as

$$\begin{aligned} \tilde{\theta}_k &= \hat{\theta}_k - \theta, \\ V_k &= \tilde{\theta}_k^T P_k^{-1} \tilde{\theta}_k. \end{aligned}$$

From (4.9), (4.16) and (4.25), we obtain

$$\begin{aligned} \tilde{\theta}_k &= \tilde{\theta}_{k-1} + r_v^{-1} P_k \varphi_k e_k \\ &= \tilde{\theta}_{k-1} + r_v^{-1} P_{k-1} \varphi_k \eta_k. \end{aligned} \quad (4.27)$$

With (4.17) and (4.27), V_k can be further evaluated as

$$V_k = V_{k-1} + r_v^{-1} \gamma_k (\varphi_k^T \tilde{\theta}_k)^2 + 2r_v^{-1} \varphi_k^T \tilde{\theta}_k \eta_k - r_v^{-2} \varphi_k^T P_k \varphi_k (1 - r_v^{-1} \varphi_k^T P_k \varphi_k) e_k^2.$$

Let us define

$$\begin{aligned} \tilde{u}_k &= -\varphi_k^T \tilde{\theta}_k, \\ \tilde{y}_k &= \frac{1}{2} \varphi_k^T \tilde{\theta}_k + (\bar{\eta}_k - v_k). \end{aligned}$$

Then we have

$$\begin{aligned} V_k &= V_{k-1} - 2r_v^{-1} \gamma_k \tilde{u}_k \tilde{y}_k + 2r_v^{-1} \gamma_k \varphi_k^T \tilde{\theta}_k v_k - r_v^{-2} \varphi_k^T P_k \varphi_k (1 - r_v^{-1} \varphi_k^T P_k \varphi_k) e_k^2 \\ &= V_{k-1} - 2r_v^{-1} \gamma_k \tilde{u}_k \tilde{y}_k + 2r_v^{-1} \gamma_k \varphi_k^T \tilde{\theta}_{k-1} v_k + 2r_v^{-2} \varphi_k^T P_k \varphi_k [(e_k - \gamma_k v_k)^T v_k + \gamma_k v_k^2] \\ &\quad - r_v^{-2} \varphi_k^T P_k \varphi_k (1 - r_v^{-1} \varphi_k^T P_k \varphi_k) \tau_k^2 + 2r_v^{-2} \varphi_k^T P_k \varphi_k (1 - r_v^{-1} \varphi_k^T P_k \varphi_k) \tau_k v_k \\ &\quad - r_v^{-2} \varphi_k^T P_k \varphi_k (1 - r_v^{-1} \varphi_k^T P_k \varphi_k) v_k^2 \\ &\leq V_{k-1} - 2r_v^{-1} \gamma_k \tilde{u}_k \tilde{y}_k + 2r_v^{-1} \gamma_k \varphi_k^T \tilde{\theta}_{k-1} v_k + 2r_v^{-2} \varphi_k^T P_k \varphi_k [(e_k - \gamma_k v_k)^T v_k + \gamma_k v_k^2] \\ &\quad - r_v^{-2} \varphi_k^T P_k \varphi_k (1 - r_v^{-1} \varphi_k^T P_k \varphi_k) \tau_k^2 + 2r_v^{-2} \varphi_k^T P_k \varphi_k (1 - r_v^{-1} \varphi_k^T P_k \varphi_k) \tau_k v_k. \end{aligned} \quad (4.28)$$

Note that $\varphi_k^T \tilde{\theta}_{k-1}$, $e_k - \gamma_k v_k$, $\varphi_k^T P_k \varphi_k$ and τ_k are uncorrelated with v_k and \mathcal{F}_{k-1} -measurable.

Thus taking the conditional expectation of both sides of (4.28) with respect to \mathcal{F}_{k-1} gives

$$\begin{aligned} \mathbb{E}(V_k | \mathcal{F}_{k-1}) &\leq V_{k-1} - 2r_v^{-1} \gamma \mathbb{E}(\tilde{u}_k \tilde{y}_k) + 2r_v^{-1} \gamma \mathbb{E}(\varphi_k^T P_k \varphi_k) \\ &\quad - r_v^{-2} \gamma \mathbb{E}[\varphi_k^T P_k \varphi_k (1 - r_v^{-1} \varphi_k^T P_k \varphi_k)] \bar{\tau}_k^2. \end{aligned} \quad (4.29)$$

Consider that

$$\begin{aligned}
A_z(\bar{\eta}_k - v_k) &= A_z(y_k - x_{b,k}) \\
&= B_z u_k - A_z x_{b,k} \\
&= -\varphi_k^T \tilde{\theta}_k = \tilde{u}_k.
\end{aligned}$$

Therefore, we have

$$\tilde{y}_k = \left(\frac{1}{A_z} - \frac{1}{2} \right) \tilde{u}_k.$$

In (A2), it is assumed that $\left(\frac{1}{A_z} - \frac{1}{2} \right)$ is positive real, which indicates

$$S_k := 2r_v^{-1} \sum_{i=1}^k \gamma \tilde{u}_i \tilde{y}_i \geq 0, \quad \text{a.s.} \quad (4.30)$$

Adding S_k to both sides of (4.29) yields

$$\begin{aligned}
\mathbb{E}(V_k + S_k | \mathcal{F}_{k-1}) &\leq V_{k-1} + S_{k-1} + 2r_v^{-1} \gamma \mathbb{E}(\varphi_k^T P_k \varphi_k) \\
&\quad - r_v^{-2} \gamma \mathbb{E}[\varphi_k^T P_k \varphi_k (1 - r_v^{-1} \varphi_k^T P_k \varphi_k)] \bar{\tau}_k^2.
\end{aligned} \quad (4.31)$$

Define a new sequence:

$$W_k = \frac{V_k + S_k}{(\ln \mathbb{E}|P_k^{-1}|)^c}, \quad c > 1. \quad (4.32)$$

Since $\ln \mathbb{E}|P_k^{-1}|$ is nondecreasing and $\varphi_k^T P_k \varphi_k = o(1)$, there exists a k_0 such that if $k \geq k_0$ we have

$$\begin{aligned}
\mathbb{E}(W_k | \mathcal{F}_{k-1}) &\leq \frac{V_{k-1} + S_{k-1}}{(\ln \mathbb{E}|P_k^{-1}|)^c} + \frac{2r_v^{-1} \gamma \mathbb{E}(\varphi_k^T P_k \varphi_k)}{(\ln \mathbb{E}|P_k^{-1}|)^c} \\
&\quad - \frac{r_v^{-2} \gamma \mathbb{E}[\varphi_k^T P_k \varphi_k (1 - r_v^{-1} \varphi_k^T P_k \varphi_k)] \bar{\tau}_k^2}{(\ln \mathbb{E}|P_k^{-1}|)^c} \\
&\leq W_{k-1} + \frac{2r_v^{-1} \gamma \mathbb{E}(\varphi_k^T P_k \varphi_k)}{(\ln \mathbb{E}|P_k^{-1}|)^c} \\
&\quad - \frac{r_v^{-2} \gamma \mathbb{E}(1 - r_v^{-1} \varphi_k^T P_k \varphi_k) \bar{\tau}_k^2}{(\ln \mathbb{E}|P_k^{-1}|)^c}.
\end{aligned} \quad (4.33)$$

From (4.12) we have

$$\mathbb{E}(1 - r_v^{-1} \varphi_k^T P_k \varphi_k) > 0.$$

Also note that by Lemma 4.2 the summation of the third term in (4.33) from 0 to ∞ is finite. Therefore, Theorem 3.1 is applicable, and it gives

$$\sum_{k=1}^{\infty} \frac{r_v^{-2} \gamma \mathbb{E} (1 - r_v^{-1} \varphi_k^T P_k \varphi_k) \bar{\tau}_k^2}{(\ln E|P_k^{-1}|)^c} < \infty \quad \text{a.s.} \quad (4.34)$$

Further, Lemma 4.1 indicates

$$\sum_{k=1}^{\infty} \frac{r_v^{-2} \gamma \mathbb{E} (1 - r_v^{-1} \varphi_k^T P_k \varphi_k) \bar{\tau}_k^2}{(\ln E r_k)^c} < \infty \quad \text{a.s.} \quad (4.35)$$

As $[1 - r_v^{-1} \mathbb{E} (\varphi_k^T P_k \varphi_k)]$ is positive and nondecreasing, it holds that $1 = O[1 - r_v^{-1} \mathbb{E} (\varphi_k^T P_k \varphi_k)]$.

Hence,

$$\sum_{i=1}^{\infty} \frac{\bar{\tau}_i^2}{(\ln E r_i)^c} < \infty \quad \text{a.s.} \quad (4.36)$$

Since $\lim_{k \rightarrow \infty} \ln E r_k = \infty$, then from the Kronecker lemma [31] it follows that

$$\lim_{k \rightarrow \infty} \Delta_k = 0, \quad \text{a.s.},$$

where

$$\Delta_k \triangleq \frac{1}{(\ln E r_k)^c} \sum_{i=1}^k \bar{\tau}_i^2.$$

With

$$r_k = \frac{n}{p_0} + \sum_{i=1}^k r_v^{-1} \gamma \varphi_i^T \varphi_i$$

and (4.23), we obtain

$$\begin{aligned} \frac{1}{k} \sum_{i=1}^k \bar{\tau}_i^2 &= \frac{\Delta_k}{k} O[(\ln E r_k)^c] \\ &= \frac{\Delta_k}{k} O[E(r_k)] \\ &= \frac{\Delta_k}{k} O\left(\frac{n}{p_0} + n_a \sum_{i=1}^k \mu_i \mathbb{E}^2(z_i) + n_b \sum_{i=1}^k u_i^2\right) \\ &= \Delta_k O\left(\frac{1}{k} \sum_{i=1}^k y_i^2\right) \end{aligned} \quad (4.37)$$

By (4.22) we have

$$\frac{1}{k} \sum_{i=1}^k y_i^2 = O(1) + O\left(\frac{1}{k} \sum_{i=1}^k \eta_i^2\right). \quad (4.38)$$

Substituting (4.37) into (4.38) gives

$$\frac{1}{k} \sum_{i=1}^k y_i^2 = O(1), \quad \text{a.s.},$$

which implies together with (4.37) that

$$\lim_{k \rightarrow \infty} \frac{1}{k} \sum_{i=1}^k \bar{\tau}_i^2 = 0, \quad \text{a.s.},$$

or equivalently

$$\lim_{k \rightarrow \infty} \frac{1}{k} \sum_{i=1}^k (y_{r,i} - y_i + v_i)^2 = 0, \quad \text{a.s.} \quad (4.39)$$

Since

$$\begin{aligned} \mathbb{E} \{ (y_{r,k} - y_k + v_k)^2 | \mathcal{F}_{k-1} \} &= \mathbb{E} [(y_{r,k} - y_k)^2 + 2y_{r,k}v_k - 2y_kv_k + v_k^2 | \mathcal{F}_{k-1}] \\ &= \mathbb{E} [(y_{r,k} - y_k)^2 | \mathcal{F}_{k-1}] + 0 - 2r_v + r_v \\ &= \mathbb{E} [(y_{r,k} - y_k)^2 | \mathcal{F}_{k-1}] - r_v, \quad \text{a.s.}, \end{aligned}$$

and $\gamma_k z_k = \gamma_k y_k$, we have

$$\lim_{k \rightarrow \infty} \frac{1}{k} \sum_{i=1}^k \mu_i \mathbb{E} \{ (z_i - y_{r,i})^2 | \mathcal{F}_{i-1} \} = \lim_{k \rightarrow \infty} \frac{1}{k} \sum_{i=1}^k \mu_i \mathbb{E} \{ (y_i - y_{r,i})^2 | \mathcal{F}_{i-1} \} = r_v, \quad \text{a.s.}$$

This completes the proof. \square

4.5 Numerical Examples

In this section, we give three examples to illustrate the adaptive control design scheme proposed in the previous sections.

The OE model used in the simulation is chosen as

$$y_k = \frac{b_0 + b_1 z^{-1} + b_2 z^{-2}}{1 + a_1 z^{-1} + a_2 z^{-2}} u_k + v_k,$$

which is assumed to be placed in a network environment (Fig. 4.1) with randomly missing output measurements and unknown model parameters. $\{v_k\}$ is a white noise sequence with zero mean and variance $r_v = 0.05^2$. The parameter vector $\theta = [a_1 \ a_2 \ b_0 \ b_1 \ b_2]^T$ is to be estimated. Here, true values of θ are

$$\theta = [-0.3 \ 0.6 \ 0.5 \ 0.2 \ 0.34]^T.$$

For simulation purposes, we assume that: (1) θ is unknown and initialized by ones; (2) the output measurement $\{y_k\}$ is subject to *randomly* missing when transmitted to the controller node; (3) the availability of the output measurements (y_k) at the controller node

is characterized by the probability γ ; (4) The desired output signal to be tracked is a square wave alternating between -1 and 1 with a period of 1000. Mathematically, it is given by

$$y_{r,(500i+j)} = (-1)^{i+1}, \quad i = 0, 1, 2, \dots, \quad j = 1, 2, \dots, 500.$$

In the following simulation studies, we carry out experiments for three different scenarios regarding the availability of the output measurements at the controller node and the parameter variation, and examine the control performance, respectively. According to the proposed adaptive control scheme shown in Fig. 4.2, we apply the algorithms of the missing output estimator, model parameter estimator, and the adaptive control law to the networked control system.

Example 1: $\gamma = 0.85$. In the first example, 85% of all the measurements are available at the controller node after network transmission from the sensor to the controller. The output response is shown in Fig. 4.3, from which it is observed that the output tracking performance is satisfactory. In order to take a closer observation on the model parameter estimation and output estimation, we define the relative parameter estimation error as

$$\delta_{par}\% = \frac{\|\hat{\theta}_k - \theta\|}{\|\theta\|} \times 100\%.$$

It is shown in Fig. 4.4 (solid blue curve) that $\delta_{par}\%$ is becoming smaller with k increasing. Comparison between the estimated outputs and true outputs during the time range $501 \leq t \leq 550$ is illustrated in Fig. 4.5: The dashed lines are corresponding to the time instants when data missing occurs, and the small circles on the top of the dashed lines represent the estimated outputs at these time instants. From Fig. 4.5 it can be found that the missing output estimation also exhibits good performance.

Example 2: $\gamma = 0.65$. In the second example, a worse case subject to more severe randomly missing outputs is examined: Only 65% of all the measurements are available at the controller node. The output response is shown in Fig. 4.6. Even though the available output measurements are more scarce than those in Example 1, it is still observed that the output is tracking the desired signal with satisfactory performance. The relative parameter estimation error, $\delta_{par}\%$, is shown in Fig. 4.4 (dashed red curve). Clearly, it is decreasing when k is increasing. The estimated outputs and the true outputs are illustrated in Fig. 4.7, from which we can see good output estimation performance.

For the comparison purpose, the relative parameter estimation errors of these two examples are shown in Figure 4.4. We can see that the parameter estimation performance

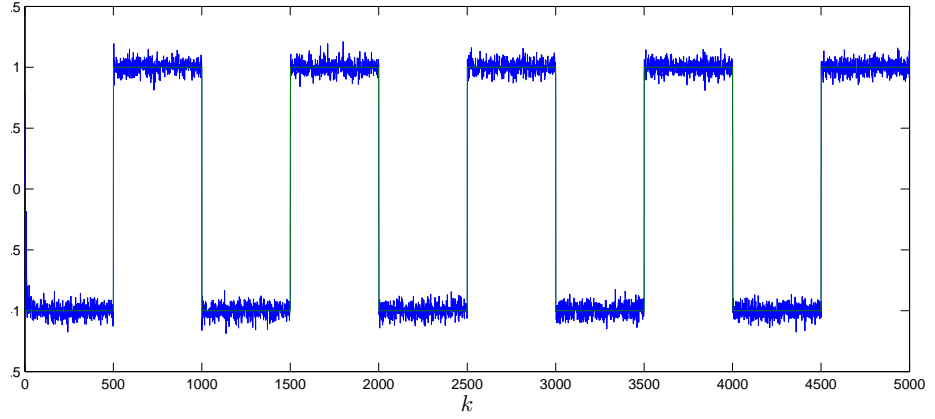


Figure 4.3: Example 1: Output response when $\gamma = 0.85$.

when $\gamma = 0.85$ is better than that when $\gamma = 0.65$. It is no doubt that the estimation performance largely depends on data completeness that is characterized by γ .

Example 3: Output tracking performance subject to parameter variation. In practice, the model parameters may vary during the course of operation due to the change of load, external disturbance, noise, and so on. Hence, it is also paramount to explore the robustness of the designed controller against the influence of parameter variation. In this example, we assume that at $k = 2500$, model parameters are all increased by 50%. The output response is shown in Fig. 4.8. It can be seen that: At $k = 2500$, the output response has a big overshoot because of the parameter variation; however, the adaptive control scheme quickly forces the system output to track the desired signal again.

Observing Figs 4.3, 4.6, and 4.8 in three examples, we notice that the tracking error and oscillation still exist. This is mainly due to (1) the missing output measurements, and, (2) the relatively high noise-signal ratio (around 25%). On the other hand, it is desirable to develop new control schemes to further improve the control performance for networked systems subject to limited feedback information, which is worth to do extensive research.

4.6 Summary

In this chapter, we have studied the problem of adaptive control for systems with SISO OE models placed in a network environment subject to unknown model parameters and

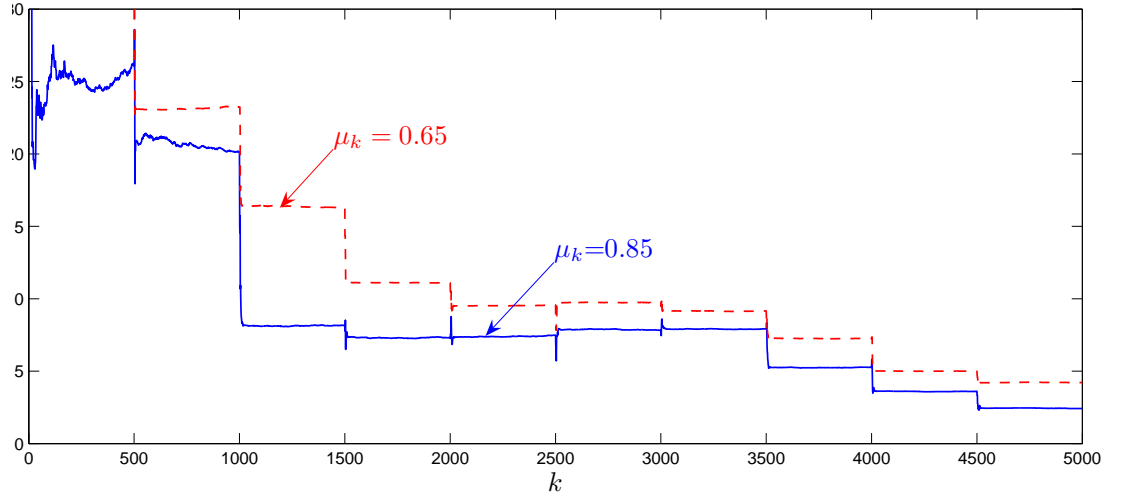


Figure 4.4: Comparison of relative parameter estimation errors for Example 1 and Example 2: Blue solid line for Example 1; red dashed line for Example 2.

randomly missing output measurements. The missing output estimator, Kalman filter based model parameter estimator, and adaptive controller have been designed to achieve output tracking. Convergence performance of the proposed algorithms is analyzed under the stochastic framework. Simulation examples verify the proposed methods. It is worth mentioning that the proposed scheme is developed for SISO systems in this work, and the extension to multi-input-multi-output (MIMO) systems is a subject worth further researching.

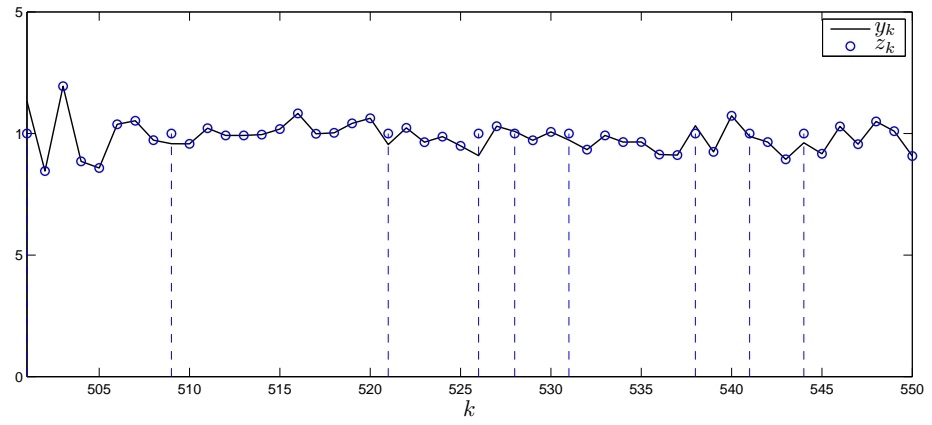


Figure 4.5: Example 1: Comparison between estimated and true outputs when $\gamma = 0.85$ (The dashed line represents output missing).

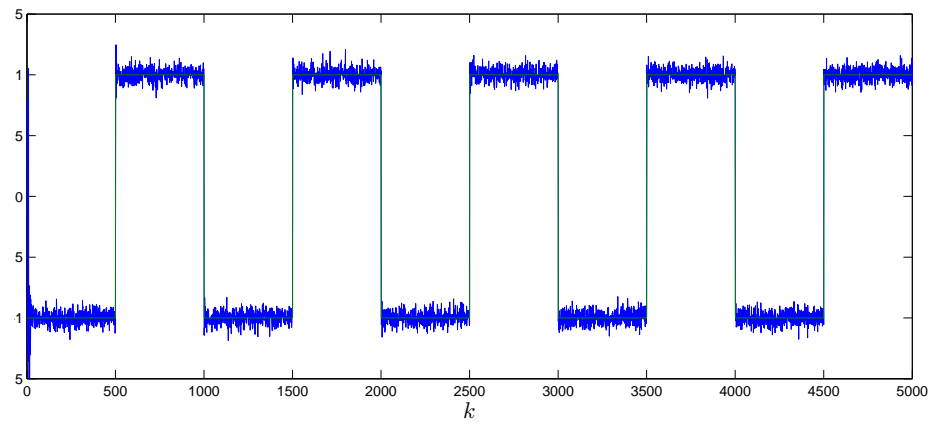


Figure 4.6: Example 2: Output response.

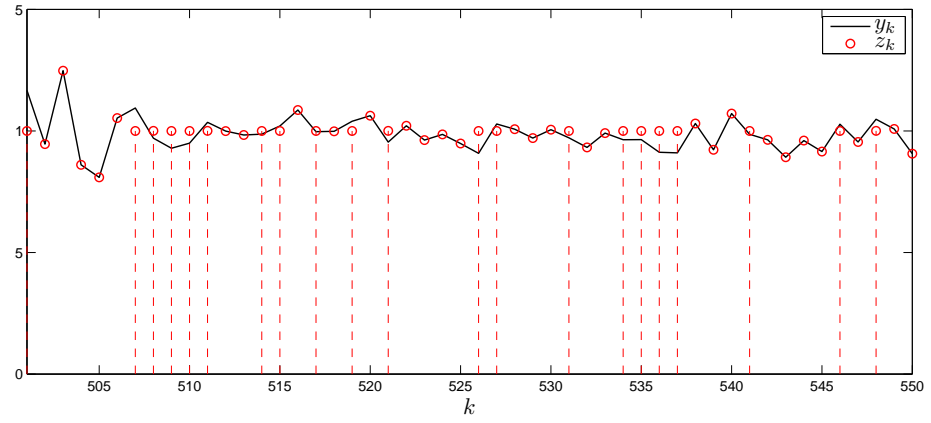


Figure 4.7: Example 2: Comparison between estimated and true outputs when $\gamma = 0.65$ (The dashed line represents output missing).

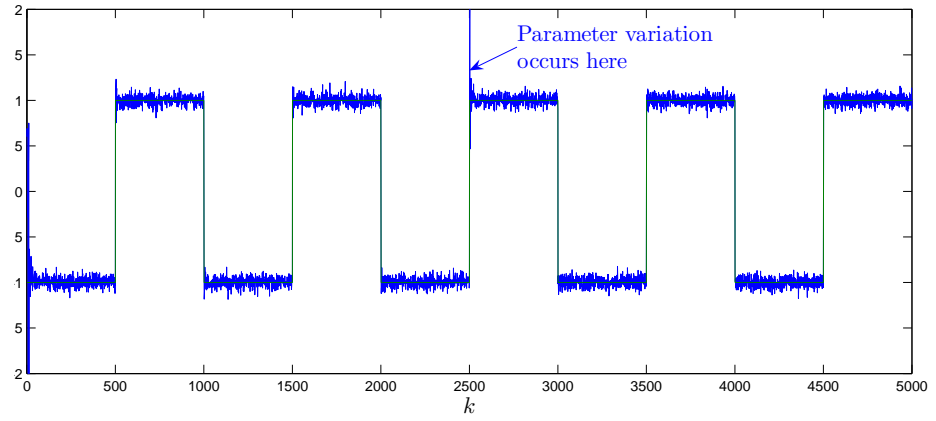


Figure 4.8: Example 3: Output response subject to parameter variation: At time $k = 2500$, all parameters are increased by 50%.

CHAPTER 5

CONCLUSIONS AND FUTURE WORK

5.1 Conclusions

This thesis explores three problems in the field of NCSs – state estimation, system identification and adaptive control. In the setting of NCSs, traditional methods addressing such problems are not directly applicable. The reason is that NCSs are inevitably subject to network-induced time delays and packet losses, which contribute to reduced information availability. Since the network factors are unavoidable, it is natural to incorporate them into the design.

In Chapter 2, we discussed the problem of state estimation in a network environment. In fact, the main challenge comes from unknown input missing, so we approached the problem using the strategy of simultaneous input and state estimation. We first started from optimality analysis, aiming at minimizing the square estimation error and error variance. However, it was noted that the algorithm was not numerically stable, so the design procedures were slightly modified. The obtained sub-optimal algorithm was then analyzed in detail to derive convergence properties. This was done by studying the solution convergence of a Riccati-like equation. We found that, if certain conditions are satisfied, the algorithm is convergent and the estimation error is upperbounded. This conclusion is worth noting, because convergence analysis, due to its difficulties, has been seldom studied in previous similar research.

In Chapter 3, we studied system identification for networked systems. The network was assumed to be based on TCP protocol, which has a loss-retransmission mechanism. We modeled the random input and output missing as Bernoulli processes. The input missing was remedied by the TCP protocol, and the output missing was handled by a designed output estimator. Furthermore, we modified the classical Kalman filter accordingly to perform recursive parameter identification. The convergence properties were analyzed

in a stochastic framework. It was found that the algorithm proposed has guaranteed convergence under some assumptions.

In Chapter 4, we focused on the problem of adaptive control for networked systems. Since NCSs run under randomly changing network conditions, adaptive control is a desirable choice for NCSs. It was supposed that the controller and the plant are collocated for ease of analysis. We studied Kalman filter based recursive plant parameter identification with missing output measurements in the first place. The identification algorithm proposed looked similar to that in Chapter 3 but was different for a different NCS setup was considered. Next, we developed the model reference adaptive control law. Finally, we analyzed convergence properties of the control law under the stochastic framework as well.

5.2 Future Work

Our research has thus far explored several significant problems and achieved meaningful results. The exploration is still at its initial stage whereas NCS research is a vast world with many areas worth further studying. It is believed that the challenges addressed in this thesis need to be examined in future research.

We observe that Chapter 2 considered only unknown input missing. In fact, output measurements are also subject to missing. Thus a problem closer to realistic setting is simultaneous input and state estimation with missing output measurements. However, this problem is very complex and our preliminary studies show that the unbiasedness of input estimation cannot be achieved when output missing occurs. Further investigation is required to tackle this problem.

Chapter 3 discussed system identification in a TCP-based network environment. Yet how can parameter identification be carried out under other protocols such as UDP? The UDP protocol employs a loss-no-retransmission mechanism, so identification becomes more challenging. Research is undergoing — we note that minimum component analysis (MCA) can be used promisingly to approach this problem. Hopefully an algorithm will be developed in near future.

In Chapter 4, the plant and controller were assumed to be collocated but linked by a network and an adaptive control law was developed. Yet how about if the plant and controller are distributed? And how to develop adaptive control laws under different pro-

ocols? These questions still necessitate further research.

REFERENCES

- [1] P. Albertos, R. Sanchis, and A. Sala. Output prediction under scarce data operation: control applications. *Automatica*, 35(10), 1999.
- [2] B. Anderson and J. Moore. Convergence properties of Riccati equation solutions. *IEEE Trans. Automat. Contr.*, 13:732–733, 1968.
- [3] B. Azimi-Sadjadi. Stability of networked control systems in the presence of packet losses. *Proc. IEEE Conf. Decision Control*, 1:676–681, 2003.
- [4] L. Cao and H. M. Schwartz. Exponential convergence of the Kalman filter based parameter estimation algorithm. *Int. J. Adapt. Contr. & Signal Processing*, 17(10), 2003.
- [5] S.W. Chan, G.C. Goodwin, and K.S. Sin. Convergence properties of the Riccati difference in optimal filtering and nonstabilizable systems. *IEEE Trans. Automat. Contr.*, 29:110–118, 1984.
- [6] C. D. Charalambous and A. Logothetis. Maximum likelihood parameter estimation from incomplete data via the sensitivity equations: the continuous-time case. *IEEE Trans. Automat. Contr.*, 45(5):928–934, 2000.
- [7] C.T. Chen. *Linear System Theory and Design*. Oxford University Press, New York, 3rd edition, 1999.
- [8] H. F. Chen and L. Guo. *Identification and Stochastic Adaptive Control*. Birkhäuser, Boston, 1991.
- [9] J.-M. Chen and B.-S. Chen. System parameter identification with input/output noisy data and missing measurements. *IEEE Trans. Signal Processing*, 48(6):1548–1558, 2000.
- [10] T. Chen and B. Francis. *Optimal Sampled-data Control Systems*. Springer, London, 1995.
- [11] M. Y. Chow and Y. Tipsuwan. Network based control systems: a tutorial. *Proc. IECON01: 27th Ann. Conf. IEEE Industrial Electronics Soc.*, 3:1593–1602, 2001.
- [12] M. Corless and J. Tu. State and input estimation for a class of uncertain systems. *Automatica*, 34:757–764, 1998.
- [13] M. Darouach and M. Zasadzinski. Unbiased minimum variance estimation for systems with unknown exogenous inputs. *Automatica*, 33:717–719, 1997.
- [14] M. Darouach, M. Zasadzinski, and M. Boutayeb. Extension of minimum variance estimation for systems with unknown inputs. *Automatica*, 39:867–876, 2003.

- [15] M. Darouach, M. Zasadzinski, and S.J. Xu. Full-order observers for linear systems with unknown inputs. *IEEE Trans. Automat. Contr.*, 39:606–609, 1994.
- [16] F. Ding and T. Chen. Parameter identification and intersample output estimation of a class of dual-rate systems. *Proc. IEEE Conf. Decision Control*, 5:5555–5560, 2003.
- [17] F. Ding and T. Chen. Combined parameter and output estimation of dual-rate systems using an auxiliary model. *Automatica*, 40(10):1739–1748, 2004.
- [18] F. Ding and T. Chen. Identification of dual-rate systems based on finite impulse response models. *Int. J. Adapt. Contr. & Signal Processing*, 18(7):589–598, 2004.
- [19] F. Ding, T. Chen, and Z. Iwai. Adaptive digital control of Hammerstein nonlinear systems with limited output sampling. *SIAM J. Contr. Optim.*, 45(6):2257–2276, 2006.
- [20] Feng Ding, Peter X. Liu, and Huizhong Yang. Parameter identification and inter-sample output estimation for dual-rate systems. *IEEE Trans. Syst., Man, Cybern. A*, 38(4):966–975, 2008.
- [21] G-R. Duan and R.J. Patton. Robust fault detection using luenberger-type unknown input observers-a parametric approach. *Int. J. of Syst. Sci.*, 32:533–540, 2001.
- [22] M. Epstein, L. Shi, and R.M. Murray. An estimation algorithm for a class of networked control systems using UDP-like communication schemes. *Proc. IEEE Conf. Decision Control*, pages 5597–5603, 2006.
- [23] T. Floquet, C. Edwards, and S.K. Spurgeon. On sliding mode observers for systems with unknown inputs. *Int. J. Adapt. Contr. & Signal Proc.*, 21:638–656, 2007.
- [24] M.E.M.B. Gaid, A. Çela, and Y. Hamam. Optimal integrated control and scheduling of networked control systems with communication constraints: Application to a car suspension system. *Proc. Amer. Control Conf.*, 14(4):776–787, 2006.
- [25] H. Gao and T. Chen. \mathcal{H}_∞ estimation for uncertain systems with limited communication capacity. *IEEE Trans. Automat. Contr.*, 52(11):2070–2084, 2007.
- [26] H. Gao and T. Chen. New results on stability of discrete-time systems with time-varying state delay. *IEEE Trans. Automat. Contr.*, 52(2):328–334, 2007.
- [27] H. Gao, T. Chen, and T. Y. Chai. Passivity and passification for networked control systems. *SIAM J. Contr. Optim.*, 46(4):1299–1322, 2007.
- [28] H. Gao, T. Chen, and J. Lam. A new delay system approach to network-based control. *Automatica*, 44(1):39–52, 2008.
- [29] S. Gillijns and B. De Moor. Unbiased minimum-variance input and state estimation for linear discrete-time systems. *Automatica*, 43:111–116, 2007.
- [30] S. Gillijns and B. De Moor. Unbiased minimum-variance input and state estimation for linear discrete-time systems with direct feedthrough. *Automatica*, 43:934–937, 2007.

- [31] G. C. Goodwin and K. S. Sin. *Adaptive Filtering, Prediction and Control*. Prentice Hall, Englewood Cliffs, 1984.
- [32] L. Guo. Estimating time-varying parameters by the kalman filter based algorithm: stability and convergence. *IEEE Trans. Automat. Contr.*, 35(2):141–147, 1990.
- [33] Q.P. Ha and H. Trinh. State and input simultaneous estimation for a class of nonlinear systems. *Automatica*, 40:1779–1785, 2004.
- [34] J. P. Hespanha, P. Naghshtabrizi, and Y. Xu. A survey of recent results in networked control systems. *Proc. of the IEEE*, 95(1):138–162, 2007.
- [35] R. A. Horn and C. R. Johnson. *Topics in Matrix Analysis*. Cambridge University Press, Cambridge, 1991.
- [36] M. Hou and R.J. Patton. Input observability and input reconstruction. *Automatica*, 34:789–794, 1998.
- [37] D. Hristu-Varsakelis. Stabilization of networked control systems with access constraints and delays. *Proc. IEEE Conf. Decision Control*, 1:1123–1128, 2006.
- [38] D. Hristu-Varsakelis and W. S. Levine. *Handbook of Networked and Embedded Control Systems*. Birkhäuser, 1989.
- [39] D. Hristu-Varsakelis and W. S. Levine, editors. *Handbook of Networked and Embedded Control Systems*. Birkhäuser, Boston, 2005.
- [40] D. Hristu-Varsakelis and K. Morgansen. Limited communication control. *Systems & Control Letters*, 37(4):193–205, 1999.
- [41] T. C. Hsia. *System Identification: Least Squares Methods*. Lexington Books, Lexington, 1977.
- [42] C. S. Hsieh. Robust two-stage Kalman filters for systems with unknown inputs. *IEEE Trans. Automat. Contr.*, 45:2374–2378, 2000.
- [43] Z. Huo and H. Fang. Robust \mathcal{H}_∞ filter design for networked control system with random time delays. *Proc. IEEE Conf. Engineering of Complex Computer Systems*, pages 333–340, 2005.
- [44] J. G. Ibrahim, S. R. Lipsitz, and N. Horton. Using auxiliary data for parameter estimation with non-ignorably missing outcomes. *Appl. Statist.*, 50(3):361–373, 2001.
- [45] A. J. Isaksson. Identification of ARX-models subject to missing data. *IEEE Trans. Automat. Contr.*, 38:813–819, 1993.
- [46] H. Ishii. \mathcal{H}_∞ control with limited communication and message losses. *Proc. Amer. Control Conf.*, pages 5620–5625, 2006.
- [47] R. H. Jones. Maximum likelihood fitting of ARMA models to time series with missing observations. *Technometrics*, 22(3):389–395, 1980.
- [48] W.S. Kerwin and J. L. Prince. On the optimality of recursive unbiased state estimation with unknown inputs. *Automatica*, 36:1381–1383, 2000.

- [49] P.K. Kitanidis. Unbiased minimum-variance linear state estimation. *Automatica*, 23:775–778, 1987.
- [50] E. K. Larsson and T. Söderström. Identification of continuous-time AR processes from unevenly sampled data. *Automatica*, 38(4):709–718, 2002.
- [51] H. Lee and M.J. Tahk. Generalized input-estimation technique for tracking maneuvering targets. *IEEE Trans. Aerosp. Electron. Syst.*, 35:1388–1402, 1999.
- [52] D. Li, S. L. Shah, and T. Chen. Identification of fast-rate models from multirate data. *Int. J. Control*, 74(7):680–689, 2001.
- [53] X.R. Li and V.P. Jilkov. A survey of maneuvering target tracking–Part IV: Decision-based methods. In *Proc. of SPIE Conf. on Signal and Data Processing of Small Targets*, pages 4728–4760, 2002.
- [54] G.P. Liu, J.X. Mu, D. Rees, and S. C. Chai. Design and stability analysis of networked control systems with random communication time delay using the modified MPC. *Int. J. Control*, 79(4):288–297, 2006.
- [55] N.B. Loden and J.Y. Hung. An adaptive PID controller for network based control systems. *Proc. Ann. Conf. IEEE Industrial Electronics Soc.*, pages 2445–2450, 2005.
- [56] S. Mirsaidi, G. A. Fleury, and J. Oksman. LMS-like AR modeling in the case of missing observations. *IEEE Trans. Signal Processing*, 45(6):1574–1583, 1997.
- [57] P. Naghshtabrizi and J. P. Hespanha. Designing an observer-based controller for a network control system. *Proc. IEEE Conf. Decision Control*, 1:848–853, 2005.
- [58] K. S. Narendra and A. M. Annaswamy. *Stable Adaptive Systems*. Prentice Hall, 1989.
- [59] D. Nešić and A.R. Teel. Input-output stability properties of networked control systems. *IEEE Trans. Automat. Contr.*, 49(10):1650–1667, 2004.
- [60] D. Nešić and A.R. Teel. Input-to-state stability of networked control systems. *Automatica*, 40(12):2121–2128, 2004.
- [61] S. Oh and S. Sastry. Distributed networked control system with lossy links: State estimation and stabilizing communication control. *Proc. IEEE Conf. Decision Control*, pages 1942–1947, 2006.
- [62] S. Oh and S. Sastry. Approximate estimation of distributed networked control systems. *Proc. Amer. Control Conf.*, pages 997–1002, 2007.
- [63] R. Patton, P. Frank, and R. Clark, editors. *Fault diagnosis in dynamic systems: Theory and application*. Prentice Hall, London, 1989.
- [64] R. Pintelon and J. Schoukens. Frequency domain system identification with missing data. *IEEE Trans. Automat. Contr.*, 45(2):364–369, 2000.
- [65] R. Sanchis and P. Albertos. Recursive identification under scarce measurements convergence analysis. *Automatica*, 38(3):535–544, 2002.

- [66] L. Schenato, B. Sinopoli, M. Franceschetti, K. Poolla, and S. S. Sastry. Foundations of control and estimation over lossy networks. *Proc. of the IEEE*, 95(1):163–187, 2007.
- [67] M. Sharan and S. Raman, editors. *Weather and Climate: The M.P. Singh Volume Part II*. Birkhäuser, Basel, 2005.
- [68] L. Shi, M. Epstein, and R.M. Murray. The effect of sensor health on state estimation. *Proc. IEEE Conf. Decision Control*, pages 3315–3320, 2006.
- [69] B. Sinopoli, L. Schenato, M. Franceschetti, K. Poolla, M.I. Jordan, and S.S. Sastry. Kalman filtering with intermittent observations. *IEEE Trans. Automat. Contr.*, 49:1453–1464, 2004.
- [70] B. Sinopoli, L. Schenato, M. Franceschetti, K. Poolla, and S. Sastry. An LQG optimal linear controller for control systems with packet losses. *Proc. IEEE Conf. Decision Control*, pages 458–463, 2005.
- [71] S. Sundaram and C.N. Hadjicostis. On delayed observers for linear systems with unknown inputs. *Proc. IEEE Conf. Decision Control*, 1:7210–7215, 2005.
- [72] M. Tabbara and D. Nešić. Input-to-state & input-output stability of networked control systems. *Proc. IEEE Conf. Decision Control*, pages 3321–3326, 2007.
- [73] Y. Tipsuwan and M. Y. Chow. Control methodologies in networked control systems. *Contr. Eng. Pract.*
- [74] D. M. Titterton. Recursive parameter estimation using incomplete data. *J. R. Statist. Soc. B*, 46(2):257–267, 1984.
- [75] R. Wallin, A.J. Isaksson, and L. Ljung. An iterative method for identification of ARX models from incomplete data. *Proc. IEEE Conf. Decision Control*, 1:203–208, 2000.
- [76] R. Wallin, A.J. Isaksson, and O. Noréus. Extension to ‘Output prediction under scarce data operation: control applications’. *Automatica*, 37(12):2069–2071, 2001.
- [77] G. C. Walsh, H. Ye, and L. G. Bushnell. Stability analysis of networked control systems. *IEEE Trans. Contr. Syst. Technol.*, 10(3):438–446, 2002.
- [78] D. Wang and K.Y. Lum. Adaptive unknown input observer approach for aircraft actuator fault detection and isolation. *Int. J. Adapt. Contr. & Signal Processing*, 21:31–48, 2007.
- [79] J. Wang, T. Chen, and B. Huang. Multirate sampled-data systems: computing fast-rate models. *J. Proc. Contr.*, 14(1):79–88, 2004.
- [80] W.M. Wonham. On a matrix Riccati equation of stochastic control. *SIAM J. Contr.*, 6:681–698, 1968.
- [81] W. Xie. Asymptotic stabilization by state feedback for a class of networked control systems. *Proc. IEEE Conf. Control & Automation*, pages 53–56, 2007.
- [82] Y. Xiong and M. Saif. Unknown disturbance inputs estimation based on a state functional observer design. *Int. J. of Syst. Sci.*, 39:1389–1398, 2003.

- [83] F. Yang and R.W. Wilde. Observers for linear systems with unknown inputs. *IEEE Trans. Automat. Contr.*, 33:677–681, 1988.
- [84] J.Y. Yu, Y.M. Mei, and S.M. Yu. MPC-based method dealing with vacant sampling and data rejection in networked control systems. *Proc. Int. Conf. Machine Learning & Cybernetics*, 1:574–578, 2004.
- [85] M. Yu, L. Wang, T. Chu, and F. Hao. An LMI approach to networked control systems with data packet dropout and transmission delays. *Proc. IEEE Conf. Decision Control*, 4:3545–3550, 2004.
- [86] D. Yue, Q. L. Han, and C. Peng. State feedback controller design of networked control systems. *IEEE Trans. Circuits Syst.*, 51(11):640–644, 2004.
- [87] K. Zhang, H. Huang, and J. Zhang. MPC-based control methodology in networked control systems. *Proc. Int. Conf. Machine Learning & Cybernetics*, 4247:814–820, 2006.
- [88] L. Zhang and D. Hristu-Varsakelis. Communication and control co-design for networked control systems. *Automatica*, 42(6):953–958, 2006.
- [89] L. Zhang, Y. Shi, T. Chen, and B. Huang. A new method for stabilization of networked control systems with random delays. *IEEE Trans. Automat. Contr.*, 50(8):1177–1181, 2005.
- [90] W. Zhang, M. Branicky, and S.M. Phillips. Stability of networked control systems. *IEEE Control Syst. Mag.*, 21:84–99, 2004.
- [91] X. Zhang, Y. Zheng, and G. Lu. Stochastic stability of networked control systems with network-induced delay and data dropout. *Proc. IEEE Conf. Decision Control*, pages 5006–5011, 2006.
- [92] Y. Zhang and S. Li. Networked model predictive control based on neighbourhood optimization for serially connected large-scale processes. *Proc. Int. Conf. Machine Learning & Cybernetics*, 17:37–50, 2007.
- [93] Q. Zhu, G. Lu, J. Cao, and S. Hu. Stability analysis of networked control systems with Markov delay. *Proc. IEEE Conf. Control & Automation*, pages 720–724, 2005.